E-Voting System based on Blockchain Technology

A major project report submitted in partial fulfilment of the requirement for the award of degree of

Bachelor of Technology

in

Computer Science & Engineering / Information Technology

Submitted by

Akshara Johari (201378)

Komal Dhall (201283)

Under the guidance & supervision of

Dr. Amol Vasudeva

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Waknaghat,

Solan - 173234 (India)

CERTIFICATE

This is to certify that the work which is being presented in the project report titled "**E-Voting System based on Blockchain Technology**" in partial fulfilment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by **Akshara Johari, 201378** and **Komal Dhall, 201283** during the period from August 2023 to May 2024 under the supervision of **Dr. Amol Vasudeva,** Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

Akshara Johari	Komal Dhal
(201378)	(201283)

The above statement made is correct to the best of my knowledge.

Dr. Amol Vasudeva
Assistant Professor (SG)

Computer Science & Engineering and Information Technology

Jaypee University of Information Technology, Solan.

Candidate's Declaration

We hereby declare that the work presented in this report entitled **'E-Voting System based on Blockchain Technology'** in partial fulfilment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of our own work carried out over a period from August 2023 to May 2024 under the supervision of **Dr. Amol Vasudeva** (Assistant Professor (SG), Department of Computer Science & Engineering and Information Technology).

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Student Name: Akshara Johari
Roll No.: 201378

Student Name: Komal Dhall
Roll No.: 201283

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Supervisor Name: Dr. Amol Vasudeva
Designation: Assistant Professor (SG)
Department: Department of Computer Science & Engineering and Information Technology
Dated:

ACKNOWLEDGEMENT

Firstly, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully. We are really grateful and wish our profound indebtedness to Supervisor **Dr. Amol Vasudeva, Assistant Professor (SG)**, Department of CSE Jaypee University of Information Technology, Waknaghat. Deep Knowledge & keen interest of my supervisor in the field of Blockchain to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Amol Vasudeva,** Department of CSE, for his kind help to finish our project. We would also generously welcome each one of those individuals who have helped us straightforwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, we must acknowledge with due respect the constant support and patience of my parents.

Akshara Johari
(201378)

Komal Dhall
(201283)

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Technology has transformed different elements within our societies including politics and the area of democratic elections. These include vote-rigging, fraud and general lack of trust in the systems. A blockchain-based e-voting system has provided one of the best solutions to face up such difficulties.börd: To solve these challenges, blockchain-based electronic voting system is an emerging effective way. The goal of this project is to develop an efficient e-voting infrastructure that protects voters’ data and ensures transparency in the process. Using its key features of decentralisation, immutability, and transparency, the proposed e-voting system guarantees that the voting process is intact and secure. This system is able to eliminate middlemen through the use of smart contracts to facilitate trust-free transactions among the participants. The key components of the blockchain-based e-voting system include the following: it comprises a distributed network of nodes, a consensus procedure, voter user interface and a secure backend. Once the vote is logged into the blockchain, no one cannot modify the system hence audible. Secondly, the application of cryptography ensures that the voters remain anonymous and private about them. Extensive testing and analysis will be carried out to evaluate the system’s performance and security. A series of test cases such as attempts at attack or vulnerability of the system will be implemented in order to prove its durability. User testing will also be carried out to help ensure a simple and seamless interaction and navigation system for easy user experience. The results of this project will be very important for future advances in electronic voting systems. It is worth noting that there are a number of benefits associated with using the blockchain-based approach, including improved security, transparency and operational efficiency. The system is based on decentralisation which limits any chances for tampering with the results or other forms of fraud thus creating trust with regard to the credibility of the results.

Therefore, as summarised, this project aims at developing an extremely secure and dependable blockchain-based electronic voting infrastructure. Unlike other systems where such shortcoming exists, such as being centralised and tamper prone, the proposed system overcomes these aspects through inherent blockchain characteristics like decentralisation and immutability. It can be used in the election process through introduction of trustworthiness, fairness as well as veracity into democratic elections.

CHAPTER - 1 : INTRODUCTION

1.1 INTRODUCTION

In recent years, electronic voting systems have gained popularity because they promote efficiency in the voting process as well as enhanced voter participation. Nevertheless, these systems have also received criticism because of the risks involved including hacking and vote interference. This project proposes an E-voting system that uses the latest blockchain technology for secure and unbiased voting. This project aims at developing a secure, tamper-proof, and authentic online voting scheme. Smart contract development is made possible by incorporating Truffle and Web3 frameworks. It also leverages Ganache as a blockchain simulator for testability and validation. Code of smart contract is written in Solidity, a tailor-made programming language used for writing smart contracts in the Ethereum blockchain. They use EJS, HTML, CSS, and NodeJS for front-end and back-end development respectively. This involves designing the architecture of the E-voting system, writing the smart contract in Solidity, Truffle, and Web3, testing and validating the system using Ganache, and developing both the front-end in typescript and the backend using NodeJS. This section of the report describes an assessment of the system's functionality, how it compares against conventional e-voting platforms, analysis of its security attributes, and user perspective. To conclude, this will be a project aimed at improving an old system of voting using a secure, modern online system with integrity. With blockchain technology, the voting process becomes more trusted as it is a lot harder to tamper with it. It is believed that if implemented, the system could improve voter turnout and strengthen existing democracy in India.

Lastly, blockchaining allows the electoral process to be taken to another higher level. Moreover, the immutability, transparency and reliability features of the blockchain ensures fairness in an election and creates confidence by the voters, candidates and electoral boards. The modern system will counter issues associated with electoral mismanagement, modification, and fraud hence will develop a better, reliable and responsive electoral system. This implies that it will promote democracy in the virtual world because of citizens' participation.

1.2 PROBLEM STATEMENT

It’s true that in every democracy, citizens express their opinions on national issues only during elections. Electronic devices used in voting have made the process fast but with an opportunity of cheating. For example: some of these challenges are related to transparency and manipulation in EVMs; issues with the handling of Computer cards and manipulations in punch cards etc. These challenges imply that they happen because the system is centralised. This imposes a need for a decentralised system. In these systems, once transactions get approved they become immutable and no insertion, alteration or revision is allowed. Blockchain is one approach for this. By design, data is distributed across all nodes in a blockchain database. This ensures that there is transparency as it reduces the need of having a centre which coordinates all these activities. Blockchain is also a more secure system due to its identity verification as part of preventing fraud. Malicious attacks such as double spending or even hacking records can be stopped with blockchain. This project will, therefore, look at some of the possible applications of blockchain technology in the voters’ system.

In this voting system, there are voters and one admin user. After login, they must wait to be authorised by the admin user. Authorised users will be able to cast a vote after voting is initiated by the admin. For the admin user, there will be several phases. Admin will only take action in the corresponding phase, and so on. This decentralised voting system will provide a secure and transparent voting experience for voters, making a difference compared to other online voting applications with its highly secure underlying blockchain technology. Blockchain technology’s distributed ledger technology will stop allowing anyone to delete or revert previous transactions, thus the security will increase since the database can be seen by everyone and the system can’t be manipulated. In the proposed system, authentication will be done using voter addresses. It is supposed that the admin knows which user addresses should be authenticated. Unlike traditional voting systems, operations are fast and reliable because of reduced human errors.

In conclusion, the proposed system represents a shift in the area of e-voting, using the power of blockchain to instil trust in elections. By addressing the issues of transparency, security, trust; this innovative solution redefines the democratic voting experience, safeguarding the essence of democracy in the digital age.

1.3 OBJECTIVES

This project aims at developing a reliable and safe e-voting system on the blockchain platform. This system will make sure that votes are transparent, irremovable and exact. Also, it will maintain secrecy of the vote and prevent any corruption. The project is aimed at designing a feasible and dependable e-voting mechanism that may apply to diverse election scenarios, such as local and countrywide polls.

Specifically, we plan to accomplish the following objectives:

1.3.1 To ensure that this is possible, smart contracts will be developed using Solidity making it transparent, immutable, and accurate.

1.3.2 Implement authenticated user registration and login system where voting will be verified by Aadhaar card number and hence prevent multiple or fake vote casting.

1.3.3 Develop a web application that allows voters to log into the e-voting system and vote.

1.3.4 To successfully integrate the web application with the blockchain network we will utilise Web3.js and Ganache. This integration will enable us to execute contracts and securely store voting data.

1.3.5 Our next task involves developing a vote counting system that retrieves and aggregates voting data from the blockchain. This system will be responsible for calculating the election results.

By accomplishing these objectives we aim to create an e-voting system that enhances the integrity of the election process. It will prioritise voter privacy while effectively preventing any activities. Additionally we want to design a user efficient solution that's both cost effective and easy to implement. Ultimately our project's goal is to support democracy by ensuring transparent elections.

1.4 SIGNIFICANCE AND MOTIVATION OF THE PROJECT WORK

Every democracy has to guarantee free and open elections. These include having a safe, open and decentralised election system that could effectively address these challenges and reassure the public. But traditional voting systems have several problems. Some of these problems include vote manipulations like the use of EVMs, voter bribery, ballot stuffing, hijackings etc. The blockchain technology has certain inherent features such as immutability, transparency, and decentralisation that make it ideal for building an efficient and safe e-voting methodology. This project will use the e-voting system in order to create a secure, transparent and an efficient blockchain based e-voting system. We intend to use the virtues of blockchain in establishing a secure voting system for safeguarding the authenticity of the voting process.

Besides that, the developed e-voting system aims at overcoming various problems related to the modern voting systems like voter's fraud, EVM manipulation and security issues. For instance, using blockchain enables to erase duplicates and invalid elections protecting identity of voters and establishing reliability. This project intends to support construction of a robust voting system to revive public trust in elections. We aim at conducting open and honest elections which can genuinely manifest the choice of voters.

Building an e-voting system using blockchain technology holds significant promise and motivation for several reasons:

1.4.1 SECURITY AND TRANSPARENCY: Data integrity and security are guaranteed by blockchain's decentralisation feature. Every vote is encrypted, stamped with a time-mark, and connected into a blockchain, making tampering very hard to achieve. Transparency encourages trust in the vote, which is necessary for safeguarding the authenticity of free and fair elections.

1.4.2 REDUCED FRAUD: There are so many types of frauds including ballot stuffing and tampering that can happen in the traditional election system. Using blockchain's cryptography, these risks are reduced using an unchangeable and verifiable register of votes.

1.4.3 ACCESSIBILITY AND CONVENIENCE: The use of e-voting systems makes it easier for people to vote from wherever they are at any given time without necessarily visiting polling stations. People can vote while using their gadgets anytime anywhere, thus removing the need for dedicated space allowing even those like the seniors and disabled to vote without problem.

1.4.4 COST EFFICIENCY: There is a possibility that adopting an electronic voting system based on blockchain can lower expenses associated with conventional methods. This makes it easier as it cuts on costs of paper, print, and personnel required for manual elections.

1.4.5 INCREASED VOTER ENGAGEMENT: E-Voting systems use easy-to-use interfaces and modern technology that can reach the people better. This method is more convenient for voters hence more individuals turn out to vote in future.

Nevertheless, there are challenges such as preservation of voter privacy, scalability and technical problems that should be overcome if large-scale use is to be achieved. Furthermore, people should trust the system's security and openness if they are to accept and believe it.

In summary, the importance rests on making the democratic process simpler, quicker, and more reliable by applying blockchain technology's strength and security features into designing a reliable voting option.

1.5 ORGANISATION OF PROJECT REPORT

This report is organised majorly into 6 sections and each section provides a detailed description about the project.

Chapter 1 :

1. Introduction: the introduction provides the basic layout and insights about the project work. It briefs the entire need of the proposed project work.
2. Problem Statement: a problem statement is a description of an issue or challenge that needs to be solved. It outlines the gap between the current solution and the desired solution, highlighting what needs to be solved or improved. It includes details such as the scope of the project, its impact, relevant issues or limitations, and the desired outcomes. Considering the needs of society, we have made an effort to approach the existing problems.
3. Objectives: A project objective describes the desired results of the work. We have mentioned the work. We are trying to accomplish in the section.
4. Significance and Motivation of the Project Work: here we have mentioned the contribution of our work for the society.

Chapter 2 :

1. Overview of relevant literature: The purpose of a literature review is to gain an understanding of the existing resources to a particular topic or area of study. We have referred to many research papers that are relevant to a walk in a better way.
2. Key gaps in the literature: Identifying key gaps in the literature involves highlighting areas where existing research is insufficient or incomplete. These gaps represent opportunities for further investigation and can guide future research.

Chapter 3 :

1. Requirements and Analysis: System requirement specifications describe the nature of the project, website or application. This section contains the brief knowledge about functional and non-functional requirements that are needed

to implement the project.

2. Project Design and Architecture: Project design and architecture refers to the process of planning, structuring, and organising a project to get its objectives met effectively.
3. Implementation: Implementation of the project described the detailed concept of the project. It also describes its algorithms in detail along with its code and algorithms.
4. Key Challenges: Implementation of a project can face various challenges, which may happen due to factors like resource constraints, technical complexities, external influences etc. This section highlights such challenges faced during this project's implementation.

Chapter 4 :

1. Testing Strategy: this contains the information about unit testing, validation testing. Functional testing, integration testing, user acceptance testing etc.
2. Test cases and Outcomes: test cases quotation the details about the program's test cases and its outcomes.

Chapter 5 :

1. Results: Results of a project mean to the outcomes, achievements, or deliverables produced as a result of implementing the project. This section contains the results we achieved after implementing the project.

Chapter 6 :

1. Conclusion: the conclusion is used to wrap up the discussion, reiterate the main findings, and offer insights or suggestions for future work that can be done.
2. Future Scope: Future scope refers to the potential opportunities or areas of further work that are highlighted from the findings, outcomes, or conclusions of the project. It involves identifying possibilities for future research and development based on current work done.

CHAPTER 2 : LITERATURE SURVEY

2.1 OVERVIEW OF RELEVANT LITERATURE

1. “Smart Contracts In Blockchain Technology”:

   This research paper provides a detailed overview of the current state of smart contracts in the blockchain technology, focusing on four main research questions:
   Present Condition of the Field of Study, Significance of Smart Contracts, Challenges Encountered by Smart Contracts, Future Developments of Smart Contracts. The paper analyses the theory of smart contracts in blockchain technology by reviewing papers from 2012 to 2022. It shows an increase in research output over the last few years. Smart contracts are shown as collections of tamper-proof, self-executing algorithms, offering many advantages. Despite their potential benefits, data security concerns are highlighted. Potential future developments in smart contracts are shown, including improvements in security performance, scalability etc. The paper shows the need for further research and development to address more challenges.

2. “Blockchain for electronic voting system- review and open research challenges”:

   This paper discusses the potential of blockchain technology in enhancing electronic voting systems. It starts by introducing blockchain concepts and its relevance to electronic voting. Then, it reviews existing electronic voting systems and identifies their limitations. Its focus is on how blockchain can address issues, such as lack of transparency and vulnerability to attacks. The article also highlights gaps and challenges in current research on blockchain-based electronic voting, including scalability issues and security risks. It emphasises the need for further study and suggests careful implementation, starting with small start projects. Despite the promise of blockchain, the article states that it is still an evolving technology with unresolved issues, especially concerning security and scalability.

3. “E-voting with blockchain: An E-voting protocol with decentralisation and voter privacy”:

   The paper proposes an electronic voting system based on blockchain technology, aiming to meet e-voting properties while providing decentralisation and empowering voters. It shows the challenges of implementing such a theme, particularly focusing on blockchain and smart contract limitations. It then explores existing blockchain e-voting applications and evaluates their following to these properties. The proposed protocol aims to achieve these properties while allowing voters to change their minds and cancel votes. It acknowledges the need for centralization, mainly for voter authentication, despite the desire for decentralisation. The protocol uses blockchain to store & cast ballots transparently, with each voter acting as a peer in the network responsible for maintaining consensus. It describes the voting phases like initialization, preparation, voting, and counting phases, and gives the importance of rule determination and system initialization. The paper also shows the potential of e-voting, especially among the techy youth population, and gives blockchain technology as a means to improve transparency and auditing. It also gives implementation challenges, performance measurements, and the need for improvement in blockchain technology to fully support applications like e-voting.

4. "Towards the intelligent agents for blockchain e-voting system":
   This paper was published in EUSPN (2018). It explains the interest in electronic voting system solutions to the potential issues of voting such as authentication, privacy, and data integrity. The paper presents the concept of ABVS(Auditable Blockchain Voting System). It consists of three main parts: super-node, trusted nodes, and polling stations. The paper proposes the use of intelligent agents in the ABVS e-voting system to increase security and efficiency. It introduces two types of agents: authorisation-configuration agent and voting agents. These agents facilitate the authorization, configuration, and transmission of votes between polling stations and trusted nodes, henceforth increasing the security and reliability of the e-voting system.

5. "Blockchain-Based E-Voting System":
   The paper discusses the design and implementation of a blockchain-based electronic voting system. It identifies key requirements for an e-voting system, such as preventing double voting, making sure of secure authentication, maintaining voter privacy, giving transparency, preventing vote tampering, and avoiding centralised control over election outcomes. The paper proposes a solution that uses smart contracts to facilitate secure and transparent elections. It outlines the setup of a Proof-of-Authority (POA) blockchain using Go-Ethereum, where each voting district

is represented by a node. The blockchain ensures immutability and security. The proposed system has three main roles: election administrators, voters, and district nodes. Election administrators create and manage elections, while voters authenticate themselves, cast votes, and verify their authentication after the election. District nodes handle the connection between voters and the blockchain, ensuring the integrity of the voting. The system gives voter privacy by preventing traceability from votes to voters. The paper compares different blockchain frameworks, like Exonum, Quorum, and Go-Ethereum and shows why they selected Geth for its developer-friendly features and scalability. It compared them based on consensus mechanisms, transaction rates, smart contract languages, and decentralisation. The paper emphasises the advantage of its approach, including private blockchain implementation, district-based voting, and enhancement of security against hacking.

6. “Secure Digital Voting System based on Blockchain Technology”:

   The research paper shows the growth of electronic voting systems, from traditional computer counting systems to recent technologies like Direct Recording Electronic (DRE) systems. It highlights the benefits and challenges with each approach and discusses the increasing interest in using blockchain technology for e-voting due to its cryptographic properties and end-to-end verifiability. The proposed e-voting system uses blockchain to get privacy, eligibility verification, convenience, and verifiability. The system uses fingerprinting for authentication, assigns unique voter identifiers, and generates transaction IDs for tracking the votes all the while preserving voter privacy. Implementation of the project involves developing a web-app using Java EE, MySQL for backend storage, and Multichain for blockchain integration. In conclusion, the paper highlights the potential of blockchain technology in changing voting systems by addressing major needs. Here future work involves improving blockchain resistance to double voting and increasing techniques to ensure trust in e-voting systems.

7. “Blockchain-Enabled E-Voting”:

   The article discusses the concept of blockchain-enabled e-voting and its potential to reduce concerns regarding voter access and voter fraud in traditional voting systems. It works similar to the digital currency system, where each voter receives a "wallet" containing a single "coin" representing one opportunity to vote. Votes are cast anonymously using a computer or phone, and then blockchain ensures the security, transparency, and immutability of the voting process. Several examples of implementations are given, including applications in corporate governance, community projects, city-level voting in Moscow, and national elections in countries like Sierra Leone and South Korea. These implementations demonstrate the potential of e-vote systems to increase voter participation, improve identity verification, simplify the tallying of votes, and increase transparency in the voting process. However, it also shows several challenges that need to be addressed, including public trust and acceptance, blockchain complexity, scale issues, energy use etc. Despite these challenges, the article says that BEV has the potential to transform voting systems by enhancing security, transparency, and efficiency, and reducing costs and increasing voter participation.

8. “Blockchain based E-voting recording system design”:

   This paper shows the integration of blockchain technology into the election process to address challenges like data manipulation, security, and transparency. It starts by highlighting the importance of elections in modern times and the need for secure and transparent voting systems. Blockchain is shown as a distributed, immutable, and transparent ledger that can increase the security and integrity of voting systems. The paper explains the basic principles of blockchain, including its distributed nature, consensus mechanisms, and cryptographic features. The paper outlines a design for an e-voting system based on blockchain technology. It shows the process of recording votes on the blockchain, including the making of private and public keys for each node, the verification and validation of blocks, and the creation of new blocks by a turn-by-turn system. The design aims to ensure data integrity and prevent manipulation by needing consensus from multiple nodes in the network. The paper also shows the implementation and testing of the proposed system using Python programming. It gives experimental results, including the capacity and performance of the system with different numbers of nodes. The system is found to be functional and efficient, with average processing times and storage requirements within acceptable limits.

9. “A conceptual secure Blockchain-based electronic voting system”:

   The paper discusses the evolution of the electronic voting system, focusing on new ideas in technology and security measures. It highlights good systems like those in Estonia and Norway, as well as new implementations like the New South Wales iVote system and Washington D.C.'s Digital Vote-by-Mail Service. The discussion gives the importance of security in electronic voting systems, particularly in making sure of anonymity, accuracy, and verifiability. The proposed solution introduces a blockchain-based electronic voting system designed to address security concerns. It outlines four main needs: authentication, anonymity, accuracy, and verifiability. The system uses blockchain technology to create a decentralised and tamper-proof platform for making and recording votes. Each vote is encrypted and added to a blockchain, ensuring transparency and integrity. The paper also discusses potential limitations of the system, such as the need for secure devices for voting and being unable to change a vote that is once cast. However, it concludes by highlighting the system's potential to revolutionise voting processes by giving a secure and transparent alternative to traditional methods.

10. “Bitcoin: A Peer-to-Peer Electronic Cash System”:

    The paper proposes a mega solution for electronic transactions, working to eliminate the need for trusted third parties. It gives a peer-to-peer network based on cryptographic proof, where transactions are recorded in a public history secured through a proof-of-work system. This system ensures that transactions are computationally impractical to reverse, making it safe against fraud and eliminating the reliance on intermediaries like financial institutions. By utilising a decentralised network where nodes in collection validate transactions, the proposed model establishes trust without the need for a central authority. The simplicity and robustness of the system lie in its unstructured nature, allowing nodes to join and leave the network freely while maintaining consensus through CPU computation power. This fresh approach holds the potential to revolutionise electronics by providing a secure and efficient means for direct transactions between parties.

2.2 KEY GAPS IN THE LITERATURE

Although blockchain based e-voting systems have benefits, there are still some limitations that need to be taken into account. The above proposed e-voting systems using blockchain technology reveal the following problems.

2.2.1 SCALABILITY: The scalability gap is shown by these surveys, revealing the constraints blocking the optimization of efficiency as transactions increase in volume. This leads to concern on network congestion, leading to questions about consensus mechanism algorithms and their effects on performance of the blockchain. These systems often have a limit on the number of requests they can handle in a specific time. If more requests are received it may lead to network congestions. Moreover, storing a huge number of votes on the blockchain requires additional storage and hence increases costs. Furthermore, as the nodes in a blockchain network are spread out over the entire globe, they may lead to network latency as well.

2.2.2 COMPLEXITY OF IMPLEMENTATION: Lack of complete research evaluating the real complexity of implementing blockchain-based totally e-voting systems, mainly regarding integration with existing electoral infrastructures.

2.2.3 PRIVACY CONCERNS: Inadequate exploration of capacity vulnerabilities in preserving voter anonymity within the blockchain, specifically concerning the link between public keys and voter identities.

2.2.4 SECURITY AND INTEGRITY: Limited discussion on the resilience of blockchain-based totally e-voting systems towards sophisticated cyber threats and attacks, together with DDoS attacks or manipulation of nodes within the community.

2.2.5 INABILITY TO CHANGE VOTERSs: Absence of unique analysis at the demanding situations related to the immutability of the blockchain, in particular while dealing with inaccurate or fraudulent votes.

2.2.6 CAPACITY AND TIME CONSTRAINTS: Insufficient investigation into the scalability problems of blockchain networks, which include transaction throughput barriers and capability bottlenecks all through excessive-quantity vote casting durations.

2.2.7 SPEED OF TRANSACTIONS: Lack of empirical research at the real speed and latency of transactions within a blockchain-based totally e-balloting system, especially during peak times of voting.

2.2.8 DOUBLE SPENDING PREVENTION: Inadequate exploration of potential vulnerabilities or safeguards in opposition to double spending attacks in the blockchain-primarily based voting ecosystem.

2.2.9 VOTER IDENTITY VERIFICATION: Minimal studies on powerful methods for securely dealing with voter identity verification within the blockchain, ensuring accurate validation without compromising anonymity.

Each of these areas represents an opening within the current literature that calls for further investigation and empirical look at to decorate the know-how and improvement of stable, efficient, and trustworthy e-voting structures primarily based on blockchain generation.

CHAPTER 3 : SYSTEM DEVELOPMENT

3.1 REQUIREMENTS AND ANALYSIS

3.1.1 SPECIFIC REQUIREMENTS

Specific requirements describe the external interface requirements, logical database requirements etc.

In this system following are specific requirements:

- Merkel Root
- Block Mining
- Truffle Utils

3.1.2 HARDWARE REQUIREMENTS

- Processor : Intel i5
- RAM : 4GB
- Hard Disk : 16GB

3.1.3 SOFTWARE REQUIREMENTS

- Operating System : Windows
- Backend : Javascript
- Database : MySql
- Frameworks : Blockchain
- Frintend : HTML, CSS, Javascript
- Other Requirements : HTML5 enabled browser

3.1.4 FUNCTIONAL REQUIREMENTS

- **User Authentication:** This would mean having the system demand a voter's authentication through Aadhaar or other such unique identifiers such that only those qualified to take part in the elections are allowed to do so.

- **User Roles and Access:** Users should be allowed in both categories including admin and voter. As an admin user, one needs access to the system's administrative functions, whereas a voter user needs access only for voting.

- **Blockchain Integration:** Blockchain technology should be employed in storing the voting information in a confidential manner. Smart contacts can be built and deployed by utilising Solidity, Ganache, Truffle and Web3 in this system.

- **Voting Interface:** Voting process should be simple so that voters can easily vote. It should have a screen showing the contenders along with their parties, such that the voter chooses their preferred one.

- **Vote Counting:** There will be an automated counting of votes and recording of the result in the blockchain.

- **Results Display:** A result display page should be included in the system, where it indicates how many votes were casted for various candidates and parties.

- **Security and Privacy:** There is a need to employ encryption tools together with other security safeguards to ensure the safety and confidentiality of the election procedure.

- **Error Handling:** Error detection within the system will be enabled against invalid votes and multiple voting by one person. It is also important for the user to receive appropriate messages upon making a mistake.

- **Accessibility:** All eligible voters must have access to the system, including those who are disabled.

- **Testing and Maintenance:** After creating this it should go through several tests for effectiveness and reliability. The system will also need to be kept in check so as to prevent it from getting stale or weak.

3.1.5 NON-FUNCTIONAL REQUIREMENTS

- **Security:** There is a need to design the e-voting system with strong security measures so as to prevent unauthorised entry and manipulation. There should be an inbuilt authentication, encryption, and access control to the system just for authorised persons. It is an additional point that a modern system must have the most advanced security measures as well as the latest software updates to resolve existing issues with it.

- **Scalability:** It must be possible to expand the system, so as to match up with future large scale elections. It should also possess necessary storage capacity that can hold millions of votes during peak voting periods. The system should support large volumes of votes such that in case there are many people who want to vote then it does not break down.

- **Transparency:** A transparent e-voting system must also provide opportunities for people to see how the voting results came about through various ways depending on their abilities, skills or preferences. An effective audit trail should leave no doubt as to who voted and how they voted within the system. This is because these votes cannot be changed or reversed. There are also additional requirements. For the sake of transparency and honesty of voting, the system should include the possibility of independent control over the electoral results.

- **Performance:** The e-voting system should be efficient for the voting process to run well. This can be accomplished if the system operates optimally with strong processors and memory that aid in real-time voting processing.

- **Usability:** E-voting system should be simple to operate and convenient for any voter. There should be a friendly interface for users which will involve a layout that every user can easily understand. Another thing is that it should accommodate those with a disability like visual and motor impairment.

- **Reliability:** The e-voting system needs to be highly dependable, it should be available throughout the entire course of voting. It should, therefore, be capable of handling any hardware or software failures while preserving the voting process. The system should also contain the procedures of back-up and recovery to enable continuation of the voting processes even when there is a breakdown.

- **Privacy:** The design of an e-voting system should maintain the confidentiality of the voters. At the same time, the system must protect the identities of the voters and ensure that the votes remain anonymous. To this end, the structure must also exclude anybody who may attempt to illegally access the sensitive information of the voters.

3.2 PROJECT DESIGN AND ARCHITECTURE

3.2.1 EXISTING SYSTEM

The Existing System of voting is happening manually using physical ballots and EVM's. The Voter has to himself visit the voting booths to cast Vote to a candidate so there is a lot of wastage of precious time. Due to this major factor many people especially the youth don't go out to cast their vote. This is the most worrying factor in today;'s electoral world. In a democracy each and every vote is important. There are many limitations to the traditional voting systems like -

- The existing system does not provide transparency and robustness. It is not difficult to tamper votes in today's scenario.
- Traditional systems are high In cost, because there is a need to prepare voting booths in every state and district and the cost of machines has increased a whole lot as the economy rises.
- Traditional system of voting takes a lot of time to complete the process of voting because every voter has to physically go to the stations and cast the vote. Time consuming process is also in setting up voting booths in various regions and categories. The rest of the process after voting is the counting of votes is also time consuming.

There is a pressing need for new systems for elections today. This old system can be replaced by a new online system which will put an end to the voting frauds and make the voting as well as vote counting more efficient and transparent.

3.2.2 PROPOSED SYSTEM

The current voting system requires many improvements because of the above mentioned issues. This can be achieved by replacing the ongoing system by the new system which will limit the voting frauds and make the voting as well as counting more efficient.

Our goal is to use blockchain technology to address these problems associated with traditional voting systems. Our Blockchain-enabled electronic voting solution will lower voting fraud and increase voter participation. Using a Blockchain, the most important requirements of a voting system will be satisfied:

- Anonymity: The system will prevent any communication between the votes casted by the voters and their identities.
- Authentication: Only registered voters will be allowed to vote
- Accuracy: Once a vote is cast it is permanently recorded and cannot be modified or changed in any way.
- Verifiability: The system will be verifiable such that the number of votes is accounted for.

3.2.3 BLOCKCHAIN TECHNOLOGY

The Blockchain technology can help us in implementing a system that is immutable and transparent and which cannot be hacked into. Blockchain allows the inability to change or delete any record from the block makes the Blockchain the most important technology for voting systems, Blockchain technology is supported by a distributed ledger network which has a variety of interconnected notes which are basically computer systems. Its computer system has its own copy of the ledger which is the blocked information that has the total history of all transactions in the Blockchain. There is no centralised system that controls this network. If the majority of the notes which all the computers agree and accept a transaction then only it can be added to the block chain. This network permits users to stay
Anonymous.

Blockchain technology gained popularity when the paper entitled "Bitcoin: Peer-to-Peer Electronic Cash System" was published by Satoshi Nakamoto in 2008. This paper introduced a new way of sending electronic cash in the form of bitcoins from one party to another without the need of central authority. The underlying technology that provides this peer-to-peer network is blockchain. This technology is called blockchain, because it stores

transaction data in the form of blocks that are linked together to form a chain.
Each block inside the block chain contains hashed data which is already verified and
other things that make up the entire block. It is a considerably new technology that can help to form decentralised systems, which assure the data integrity, availability, and fault tolerance.

Structure of Blockchain:

The blockchain is a decentralised and transparent ledger with the blockchain database that is shared by all the network nodes, updated by miner nodes, monitored by everyone on the network, and controlled by no one in particular. Its distributed database system keeps records of all the transactions on the blockchain across a P2P (peer-to-peer) network. In client server networks, client nodes request the data from server and server node gives data to the client nodes. In a P2P network each of the nodes can be either client or server, which means that all nodes can request for data and send data to each other. Peer-to-peer network means that all computers in the network of blockchain are connected, and each of the nodes stores a full copy of the ledger and compares it to other nodes to ensure the data is accurate.

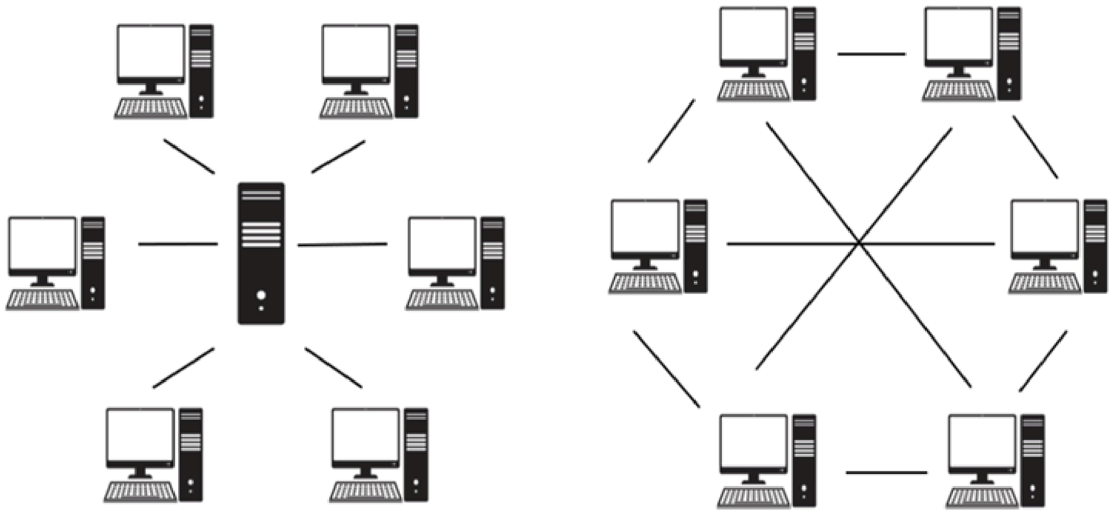


Figure 3.1. A P2P Network

Each block in the blockchain contains block header and transaction data. The block header contains metadata of the block. The metadata has six things including the version, the hash of the previous block, hash of the Merkle tree root, time stamp of the block and nonce. The main identifier of each block is the cryptographic hash it has. Each block has the hash of the block that was mined before it. This introduces immutability into the blocks. The sequence of these hashes links each block to its parent block and creates a chain going back to the first block created, known as the genesis block. The hash of the previous block field is inside the block header and hence affects the current block's hash with it. If a parent's hash changes, then the child's hash also changes. Any modification in parent causes its hash to change. This in turn changes the child's hash to change, which makes a change in the hash of the grandchild, which in turn changes the grandchild, and so on.

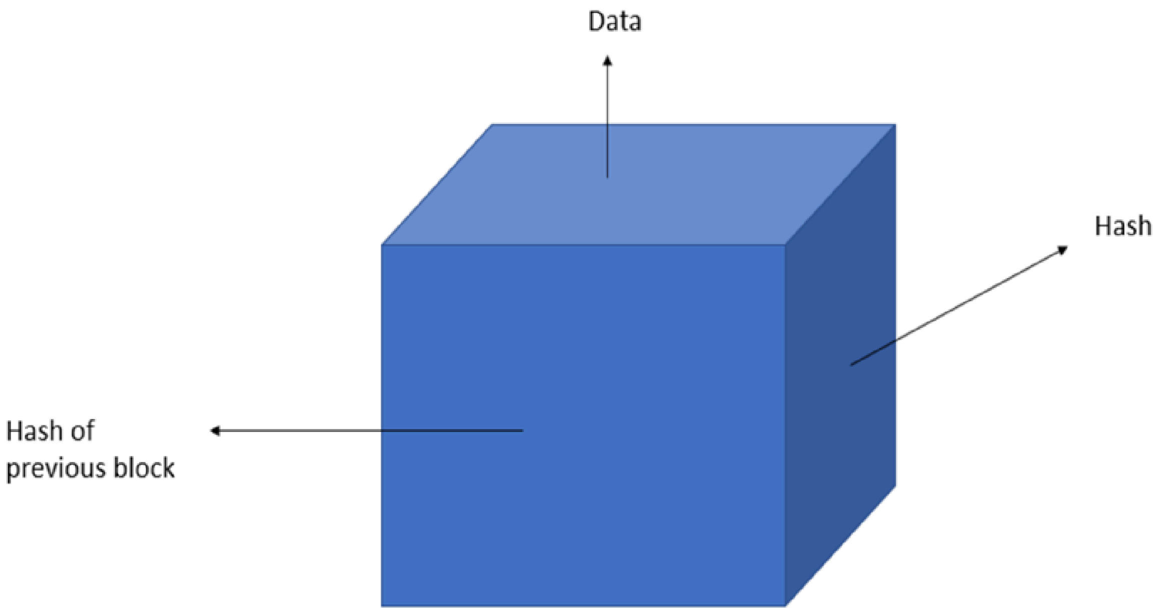


Figure 3.2. Structure of a block in blockchain

A Merkle root in blockchain is the hash of all the hashes of all the transactions that are part of a block. Nonce is a number that measures how hard it is for miners to solve the block. The nonce is a number that blockchain miner nodes are solving for. By solving this they will get cryptocurrency rewards.

Magic Number	4 bytes
Block Size	4 bytes
Version (Header)	4 bytes
Previous Block Hash (Header)	32 bytes
Merkle Root (Header)	32 bytes
Time Stamp (Header)	4 bytes
Difficulty Target (Header)	4 bytes
Nonce (Header)	4 bytes
Transaction Counter	1 to 9 bytes
Transaction	Depends on transaction size

Figure 3.3. Contents of a block

Bitcoin:

Bitcoin is a cryptocurrency that was created as a solution for "Double Spending". Double-spending is a problem that states that when sending digital currency if the currency that involves the same tender has been spent multiple times or more than once. The main reason for the double-spending problem is that digital currency can be created again very easily. Bitcoin cryptocurrency prevents double-spending by making use of security features of blockchain technology and its decentralised network of miner nodes to verify each and every transaction before they are added to the blockchain.

With blockchain as a base technology, Bitcoin provides features like security, transparency, fast transactions etc. It is not possible to control this decentralised system by any one person or entity. Users can easily make unlimited amounts of transfers and all of them can be seen by other users. But bitcoin cryptocurrency is not unlimited, only 21 million units can be produced ever, and this process will be completed by 2140. The limited supply of Bitcoin cryptocurrency has been compared with gold and has become a demanded investment opportunity.

Figure 3.4. Bitcoin

Ethereum:

Ethereum is a cryptocurrency system that was first introduced at the North American Bitcoin Conference by Ethereum founder VitalikButerin. Ethereum is a new system that aims to broaden the concept of blockchain technology and use it in more areas than the current scenario. Ethereum gives us a platform that allows for building of decentralised applications on its ethereum blockchain by running programs called smart contracts. Like other blockchain platforms like bitcoin, Ethereum uses a system called distributed ledger which is distributed among hundreds of nodes around the world. Each node participating in the Ethereum blockchain runs the ethereum software on its own machine. This big, decentralised network of nodes is what is referred to as the Ethereum Virtual Machine (EVM). Ethereum Virtual Machine makes sure of security for users by removing denial-of-service attacks (DDoS), that aim to turn off a machine or the entire network and make them completely non accessible. Through Ethereum Virtual Machine, this allows you to run applications in a distributed ledger way using the programming language called Solidity. It allows you to program transactions with certain rules that you want to specify. Every node machine participating in the ethereum network runs the EVM as part of the transaction verification process. All nodes go through the transactions listed in the block they are verifying and run the solidity code as given by the transaction within the Ethereum Virtual Machine.

Each ethereum account has a 20-byte address, and all the transactions happen between these

20-byte addresses which represent accounts. There is a public-private cryptographic key-pair which is generated for each account. While the private key is kept secret, the public key acts as the address of the ethereum account. Accounts are identified by these public keys. Smart contract accounts are managed by smart contracts written in solidity programming language. It stores the amount of ether that the account owns. In this account, the running of the program is set by any transaction or message from another ethereum account. These accounts are generated when smart contracts are deployed to the ethereum blockchain. Functions such as sending money to an account or calling a function of the smart contract are called transactions. They are digitally signed bundles of data, that are set off by an account on the Ethereum blockchain network. These messages include the receiver of the message, digital signature of the sender, amount of Ether to be transferred, STARTGAS value based on the maximum processing calculated by the ethereum virtual machine and GAS PRICE, which states the value of Ether to be paid by the sender.

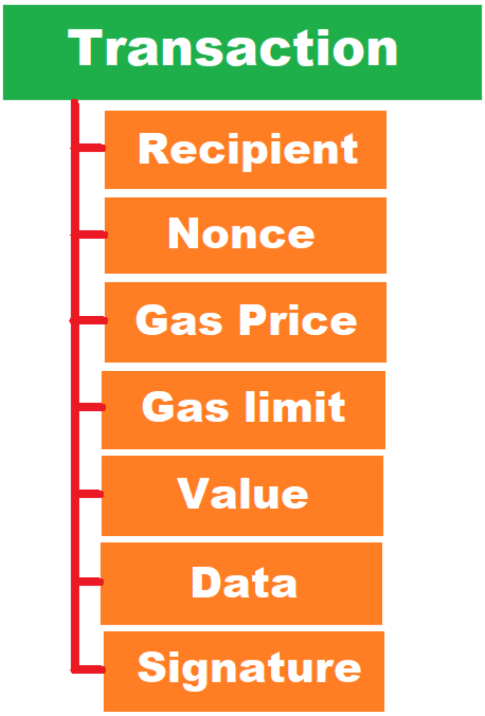


Figure 3.5. Ethereum transaction structure

Smart Contracts:

Smart Contract is a program that runs in the Ethereum Blockchain network and is executed by Ethereum Virtual Machine (EVM). Smart Contract is an immutable program, meaning once a code is written and deployed to the blockchain network, it cannot be updated or changed. Ethereum Smart Contracts are written in Solidity programming language. After the code is written and it is ready to be sent to the blockchain for deployment, the developers have the option to deploy it to Mainnet, which refers to a real network which uses real Ether. If we want to test our Smart Contract, we can deploy a Smart Contract and a test ethereum network like Ganache. Ganache does not use real Ethers. Instead it creates dummy ethers and initialises every account with 100 test ethers. A Smart Contract has a number of state variables, events, modifiers, and functions. Each call to a function that changes the state variables of the Smart Contract will be called a transaction, and each transaction will cost a specific amount of "gas". The biggest advantage of using Smart Contract is that there is no downtime as the blockchain is handled by millions of users all over the world. As long as the Ethereum blockchain network is running, the Smart Contract can be used.

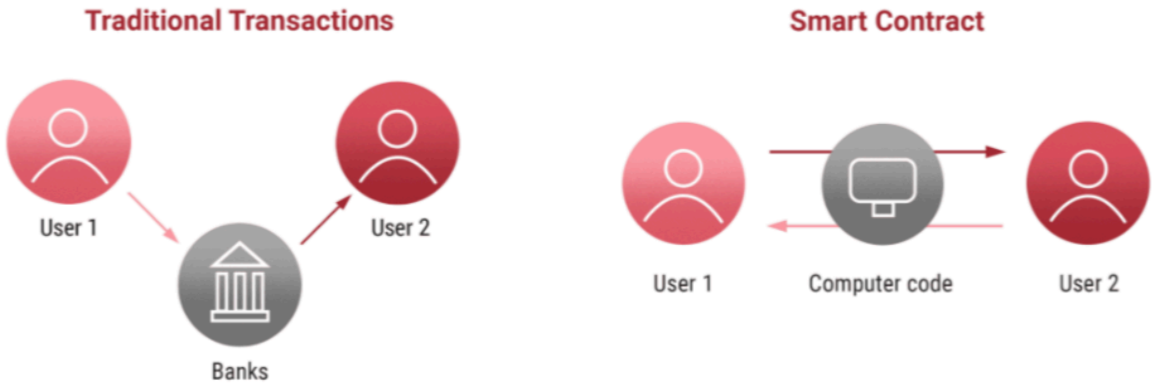


Figure 3.6. Working of smart contracts

Mining:

Mining is the process where miner nodes on the blockchain network need to perform very complicated computational calculations to be able to find a value of hash that is less than the specified nonce value. This is also called the Proof-of-Work concept. Proof-of-Work is where there is a given level of difficulty so that a miner node needs to find the nonce value when the final hash of everything inside the block and the nonce has to have more than or equal "k" number of starting zeros, and k is equivalent to the difficulty number. For example, if the difficulty is equal to 7, then the resulting hash must be 0000000asjdcdjncaah23132432dj… The difficulty is reset after some time regularly so that the average time for a block to be mined and added to the blockchain network is approximately 7 to 9 minutes. The miner who is the fastest finds the correct hash according to the nonce will be able to include their block onto the blockchain. After a block is added to the blockchain, a specific amount of reward, Bitcoin or Ether, will be awarded to that miner who has won. The main purpose of this fee is to collect other miners to join the blockchain network and help with the block validating process, thus making the blockchain network even more secure. There is also a scenario when multiple miner nodes come up with the solution simultaneously, and the blockchain has to find a way to handle that case. If three miners find the solution at the same time, the chain will split into three, and each of the sub chains needs to compete with each other to make the longest chain. This is also known as "forking". It means that in the further mining, the other miner nodes will choose one of the sub chains and mine until the longest chain is found. When this happens, the other chains will be truncated from the blockchain. This can cause to a problem when

a transaction is verified on the blockchain and then gets unverified because the chain is dropped, those transactions will be again verified quickly so this is not a big issue in the blockchain network.

SHA256 Algorithm:

SHA-256 code stands for Secure Hash Algorithm. In this hashing algorithm, hash of the data is non reversible and unique for every set of message values, it means that if we make a hash of a certain data many times with the same algorithm, the result hash would be the same, and given the hash value you cannot obtain the original data in any way. This algorithm was made by the US National Security Agency (NSA), which specialises in cryptography. SHA-256 algorithm is used in proof-of-work in cryptographic mining and address generation. SHA-256 is a member of the SHA-2 algorithm family, which is famous for high security and speed.

SHA-256 has been used by many different blockchain projects. SHA-256 is a deterministic algorithm, which means that it always produces the same specific output when the same input is there. SHA-256 is computationally light and any simple computer can perform the operation thousands of times per second. The SHA-256 algorithm is necessary and popular because it's an important part of the mining process on the blockchain network, and also many Proof of Work blockchain networks. In basic terms, the SHA 256 hash supports a Proof of Work system in which nodes compete to solve a complicated computation problem. After one node finds a solution, it sends out that solution to all other nodes on the P2P network. This validated the node's work to the other nodes that were trying to solve the same problem, because each node will verify it's work.

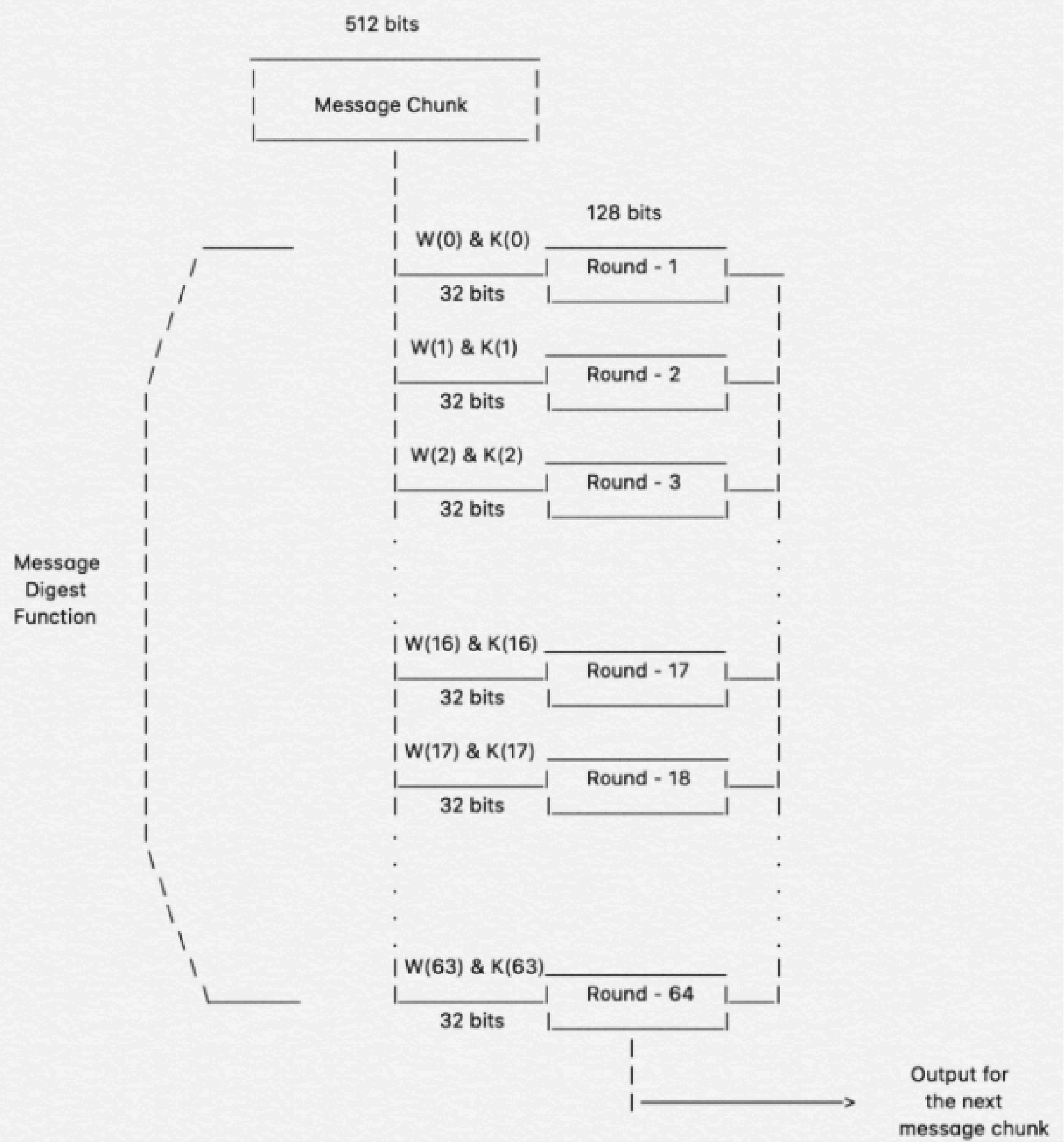


Figure 3.7. SHA256 Hashing Algorithm

Merkle Tree:

Merkle Tree is a binary tree data structure. Merkle tree uses a whole lot of data, compacts it into a simple string of characters that can certify the authenticity of the data without telling what the data was. It is guaranteed in a merkle tree that the data will not be changed by anyone. Merkle trees are made by joining two nodes one after the other until only a single hash is the output. The last hash is called the Merkle Root or Root Hash. It is made from the hash of all of the transactions, from bottom to the top. A hash function such as SHA-256 is used for hashing. The Merkle tree gives efficient and secure validation of big data. The major use of Merkle trees is to check whether data from various nodes in a P2P network are correctly transmitted and also check that other nodes in the network do not transmit blocks with compromised content.

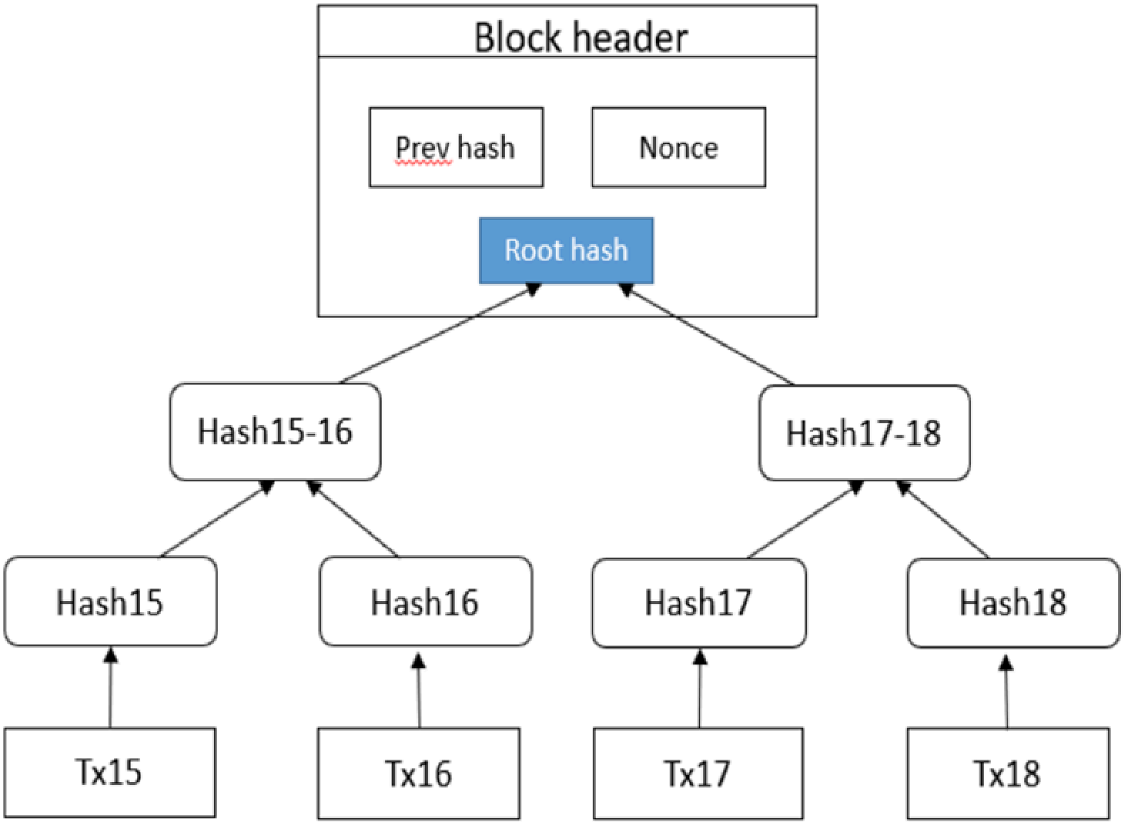


Figure 3.8. Merkle Tree

Decentralised Applications:

Decentralised Application is an application that is not dependent on a central authority of control. An application has 3 parts: Frontend, Backend, and Storage. The Frontend will mostly be hosted on a central data server. The Backend might be decentralised using Smart Contract. Smart Contract is fully stored on the blockchain network, the user will get zero downtime and the service will be up and running as long as the blockchain network is there.

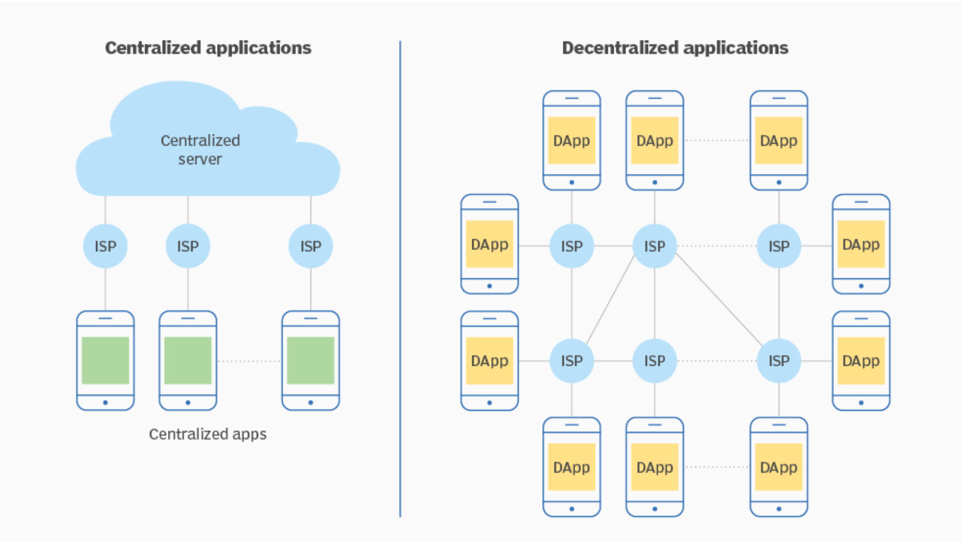


Figure 3.9. Centralised vs decentralised applications

3.2.3 DESIGN DIAGRAMS

A data flow diagram is a representation in the form of graphs that shows the information flow and the transformations that are applied as data moves from input to output stages. The common form of a data flow diagram, also called a data flow graph or a bubble chart. Data flow diagram is a low-level description of the system. The data flow diagram can be used to represent a system at any level of abstraction. Data flow diagrams may be separated into levels that represent increasing information flow and functional detail. Therefore, the Data flow diagram gives us a mechanism for functional modelling as also for information flow modelling. The diagram (3.1) describes the detailed view on the application process. It mainly shows the connection between User side architecture and the network side architecture, where the user will login using username and password. After successfully logging in and verifying identity users will vote for a candidate from the list of provided candidates. After voting the votes are given to the vote casting system in this case it is their solidity smart contract which will then hash the votes according to SHA256 algorithm and send it to the network of nodes which will then re-verify these votes and store it in their local blockchain. These votes are then sent back to the smart contracts when the command is given like for counting and showing results. The smart contact will gather all this information and will display it to the voter once the election is over. This describes our Level 0 Data flow diagram.

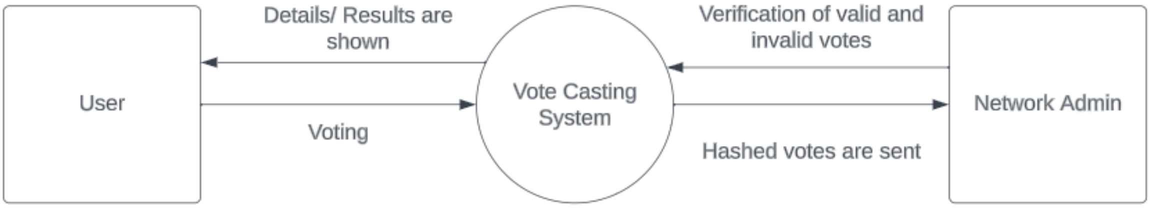

User
Details/ Results are shown
Voting
Vote Casting System
Verification of valid and invalid votes
Hashed votes are sent
Network Admin

Figure 3.10. DFD Level 0

The next diagram (3.2) is the level 1 data flow diagram and it explains the working of our project and the data flow in a more detailed manner than the above. The level 1 data flow diagram shows that when the user will login to the website, he will face an authentication process which will take place against the registered set of databases. The authentication process will take place according to an Aadhaar database system. The user will be needed to enter the details and they will be verified automatically through the verification system. Next when the user casts their vote, the vote will directly go through a smart contract system written in solidity language. The smart contract will verify if the voter is verified or not, if the voter is voting for a registered candidate or not, if the voter is above the age of 18 or not, and it will also check if or whether the voter has already voted or not, because no voter will be allowed to vote twice. After the smart contracts, the votes will be encrypted using a specific asymmetric key pair and will be digitally signed for the user who has voted and will then go to a network of nodes for the consensus process. In this process, the nodes which are computers basically will fight for who will add to note to the Blockchain. They will also verify if the vote has come from the source specified in the digital signature or not, and they will also verify the contents of the vote so that no tampering can occur. Other than this the smart contract will also be responsible for starting and ending elections and counting of votes.

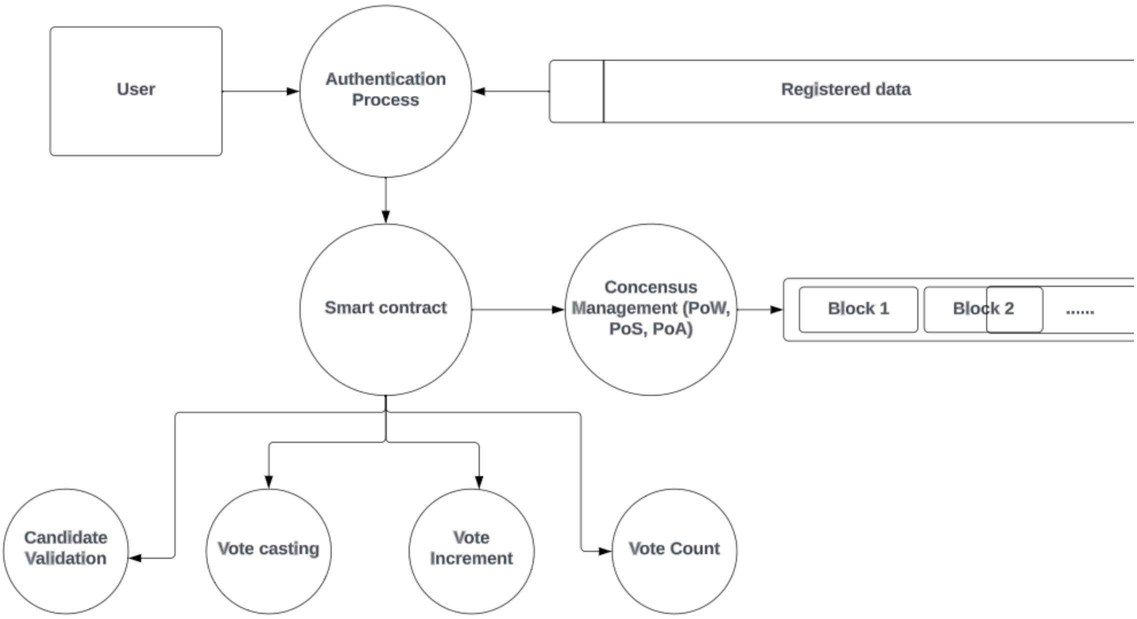


Figure 3.11. DFD Level 1

An entity–relationship diagram (ER diagram) describes relation between things of interest in a specific domain of a project. A basic ER diagram is composed of entities, which are essentially the things of interest and gives the relationships that exist between those entity types. The ER diagram is transformed into a data model that specifies a data structure that is used in a database, mostly a relational database RDB. The figure (3.3) gives an entity relation diagram between the major entities in our project along with their attributes and relationships. There are majorly 3 entities specified in this ER diagram which include the user, the candidate which the user will vote for and the actual election where the voting process will take place. The user has attributes like their Aadhaar number for verification, name, date of birth, age which is a derived attribute of the date of birth and if Voter has already voted or not. The candidate on the other hand has attributes like its candidate ID, the candidate name, the party which the candidate stands for and number of votes which the candidate has obtained. The third entity which is the election has attributes including the election ID, the start date of the election, the end date of the election and the start and end time of the election. The relation between all three of these entities is also stated in the ER diagram. The user has the relation of casting vote to the election and candidate has the relation of participating in the election election

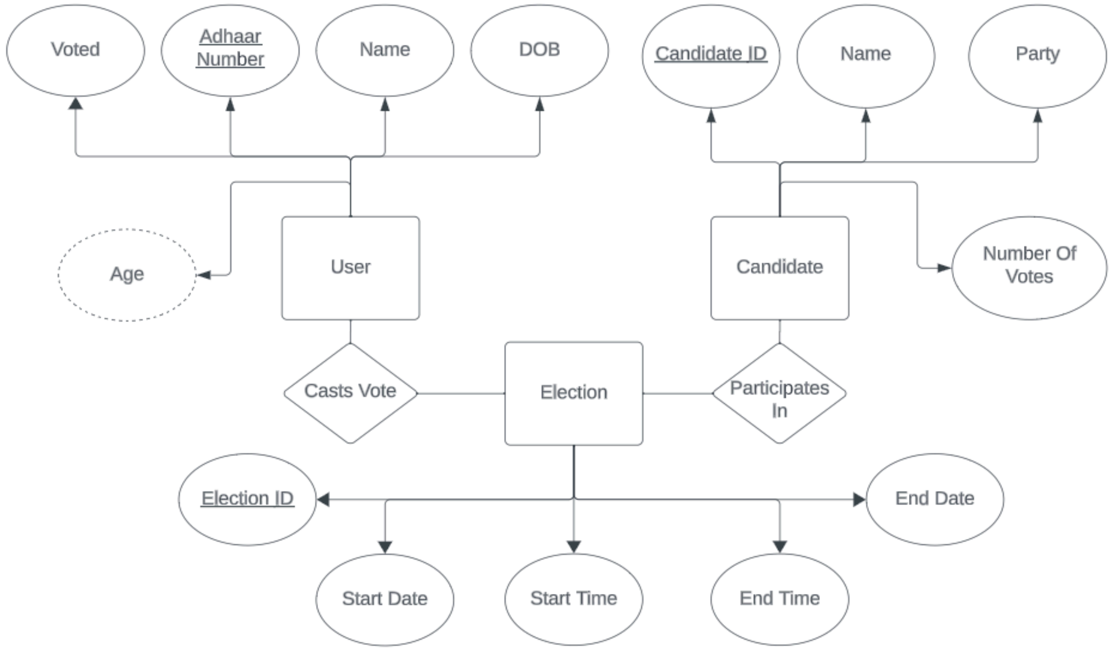


Figure 3.12. ER Diagram

Use case consists of user and admin where users are allowed to login and cast his or her vote and admin manages all the required settings like managing candidate details and authorising the right voters to cast his or her vote after the verification of the user is complete. And after the election poll ends, the admin is responsible for decoration of the result in a graphical representation.

The figure (3.13) is the use case diagram which shows how the user and server manage their processes. The processes are stated as follows. It consists of user and admin registration, candidate management, Voter management, casting of vote and declaration of result.

All of these processes are taken care of by the server and one of the users either the voter or the admin.

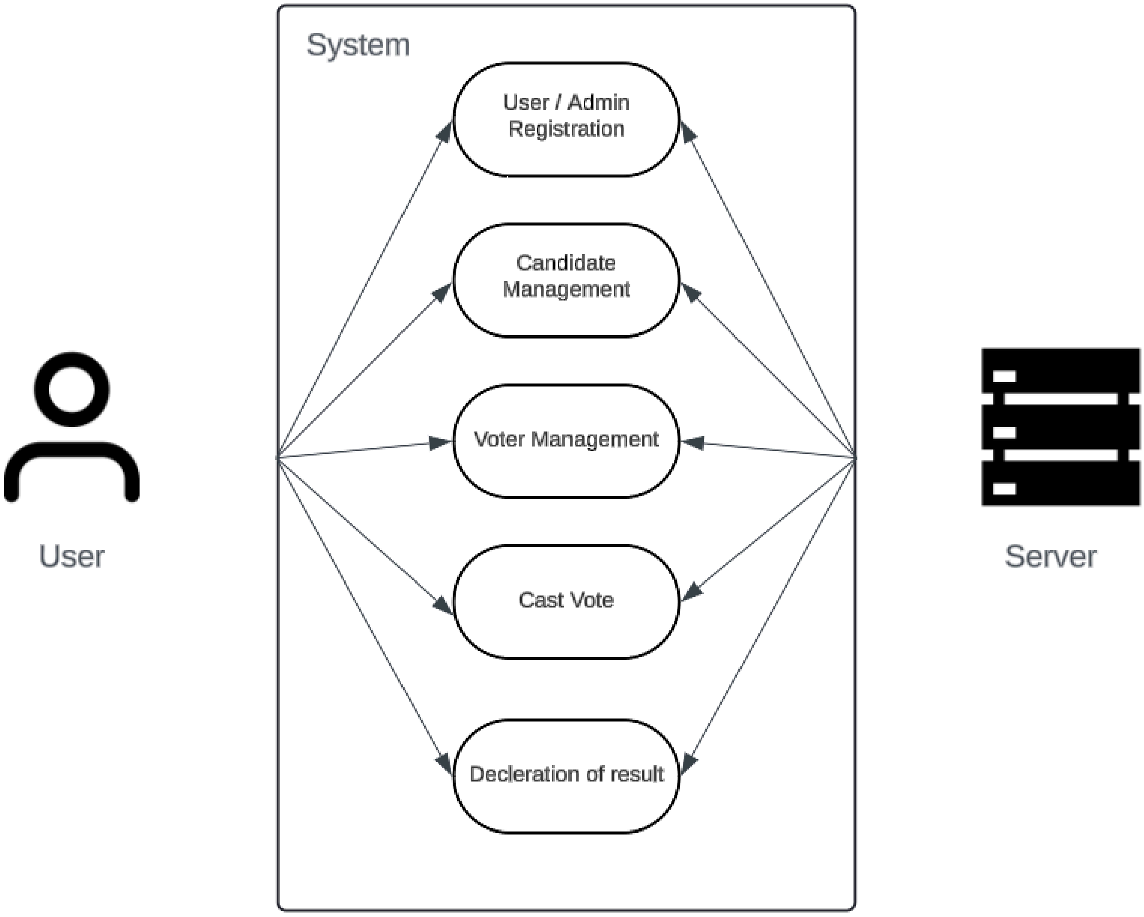


Figure 3.13. Use Case Diagram

3.3 IMPLEMENTATION

3.3.1 DEVELOPMENT AND WORKFLOW

There will be a web application for the users to interact with the project. The users will need to register themselves with an account in order to vote using their Aadhaar Number and Metamask address. After they have registered their account, they can select which candidate they want to vote for. All projects or services inside this thesis will be hosted inside a common root project directory.

The development of the decentralised application is a challenging process. The reason being the fairly new technologies used majorly the Ethereum smart contracts and the web3 library. Also a lack of documentation in their support was a major problem.

In the proposed voting system project, there are multiple voters and one admin user. Voters first have to register themselves to access the website. Authorised users will be able to cast a vote after the election is started by the admin. For the admin user, there are several steps listed below.

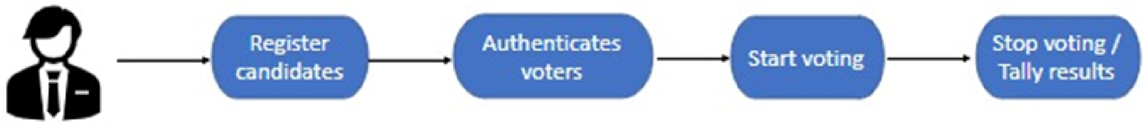


Figure 3.14. Use case for admin

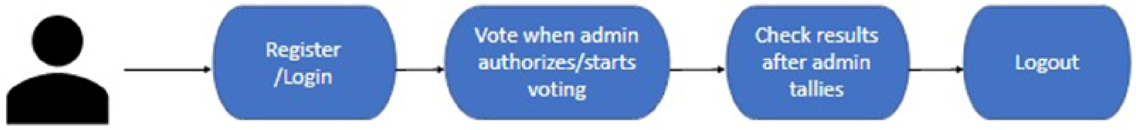

Register /Login
Vote when admin authorizes/starts voting
Check results after admin tallies
Logout

Figure 3.15. Use Case for voter

Workflow of the application is shown in the figure (3.16) below. Most of the transactions need transaction confirmation from MetaMask, as those transactions need to go through a smart contract.

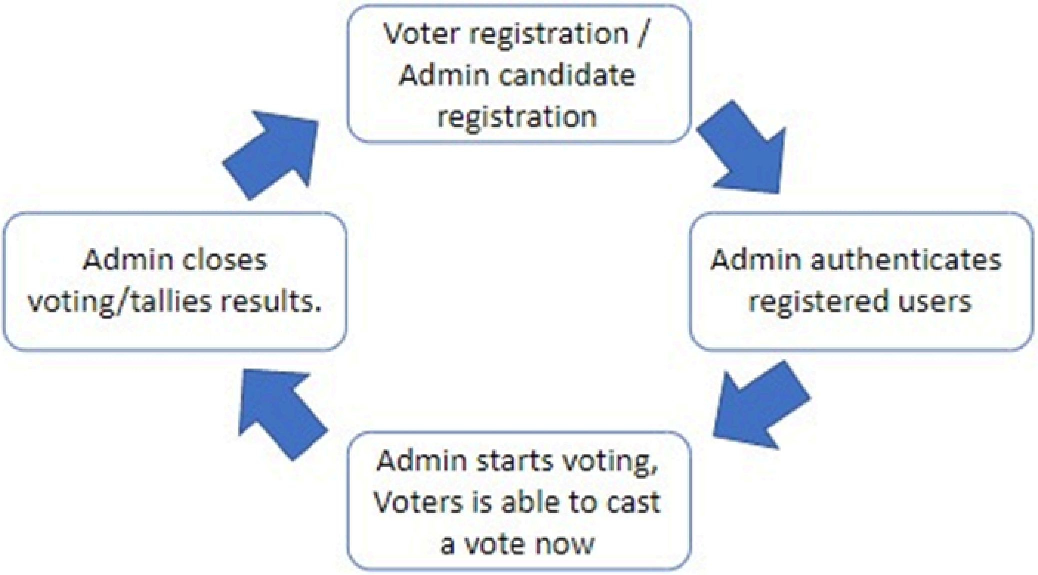
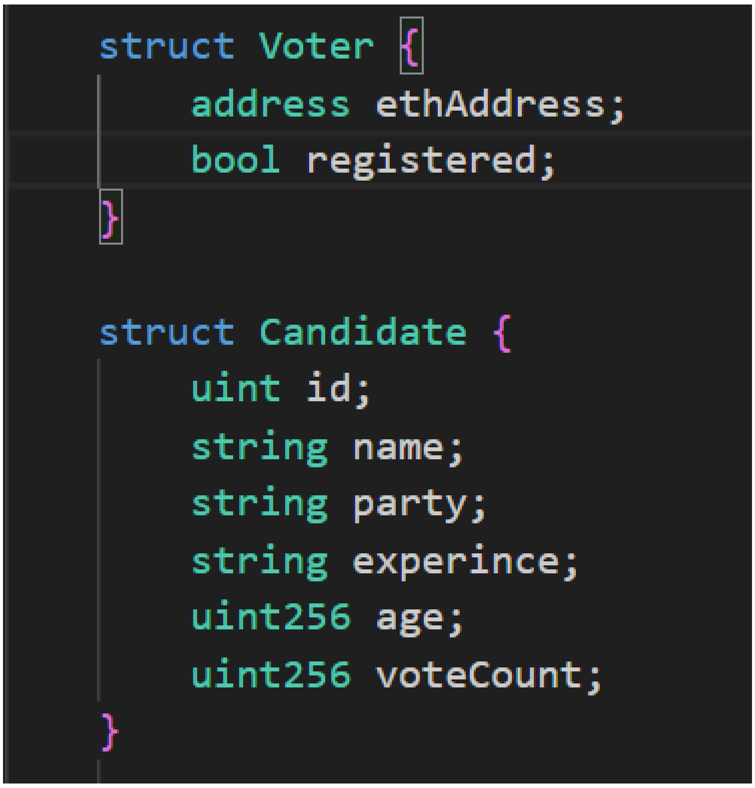

Voter registration / Admin candidate registration
Admin authenticates registered users
Admin starts voting, Voters is able to cast a vote now
Admin closes voting/tallies results.

Figure 3.16. Workflow of the website

Smart Contract:

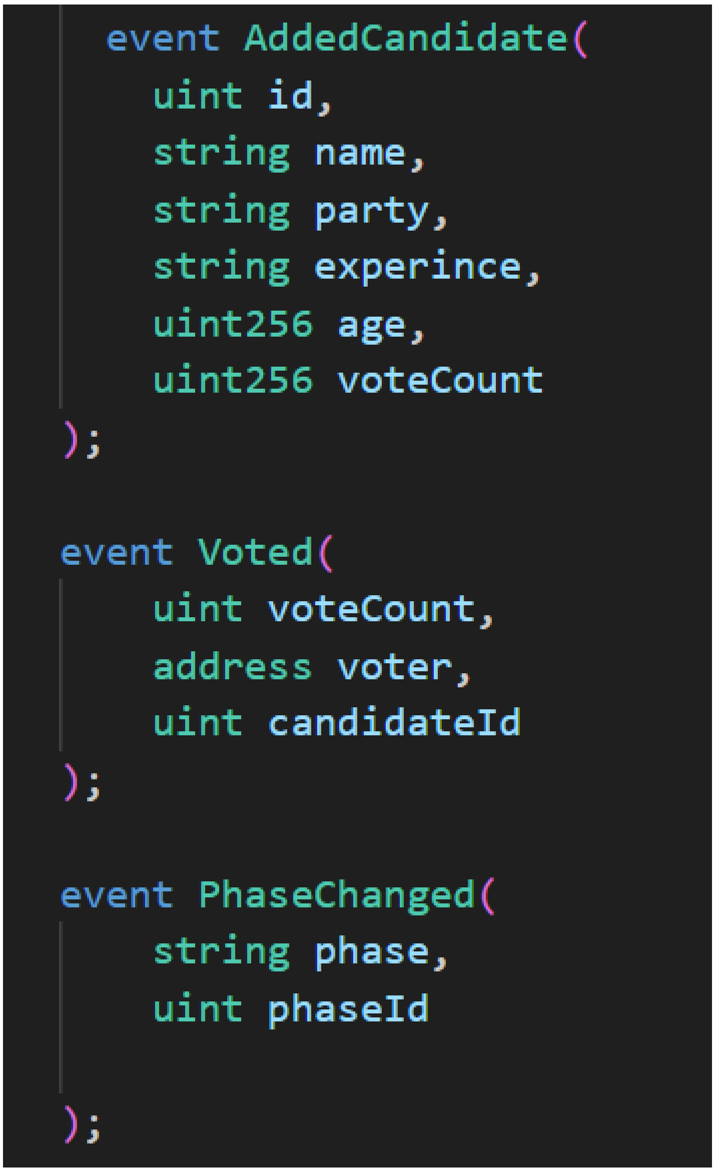
The voting smart contract (Voting.sol) written in solidity language has structures (structs) for voter and candidate. Voters have fields like if they are registered or not and their ethereum address. The candidate struct has fields like their id, name, party, experience, age and votecount. Each candidate has an id, it is set to number 1 for the first candidate and is self incrementing for every new candidate. The field called voteCount is initially set to 0, and it increases when the voters vote for that particular candidate.



```
struct Voter {
    address ethAddress;
    bool registered;
}

struct Candidate {
    uint id;
    string name;
    string party;
    string experince;
    uint256 age;
    uint256 voteCount;
}
```

Figure 3.17. Voter and Candidate struct

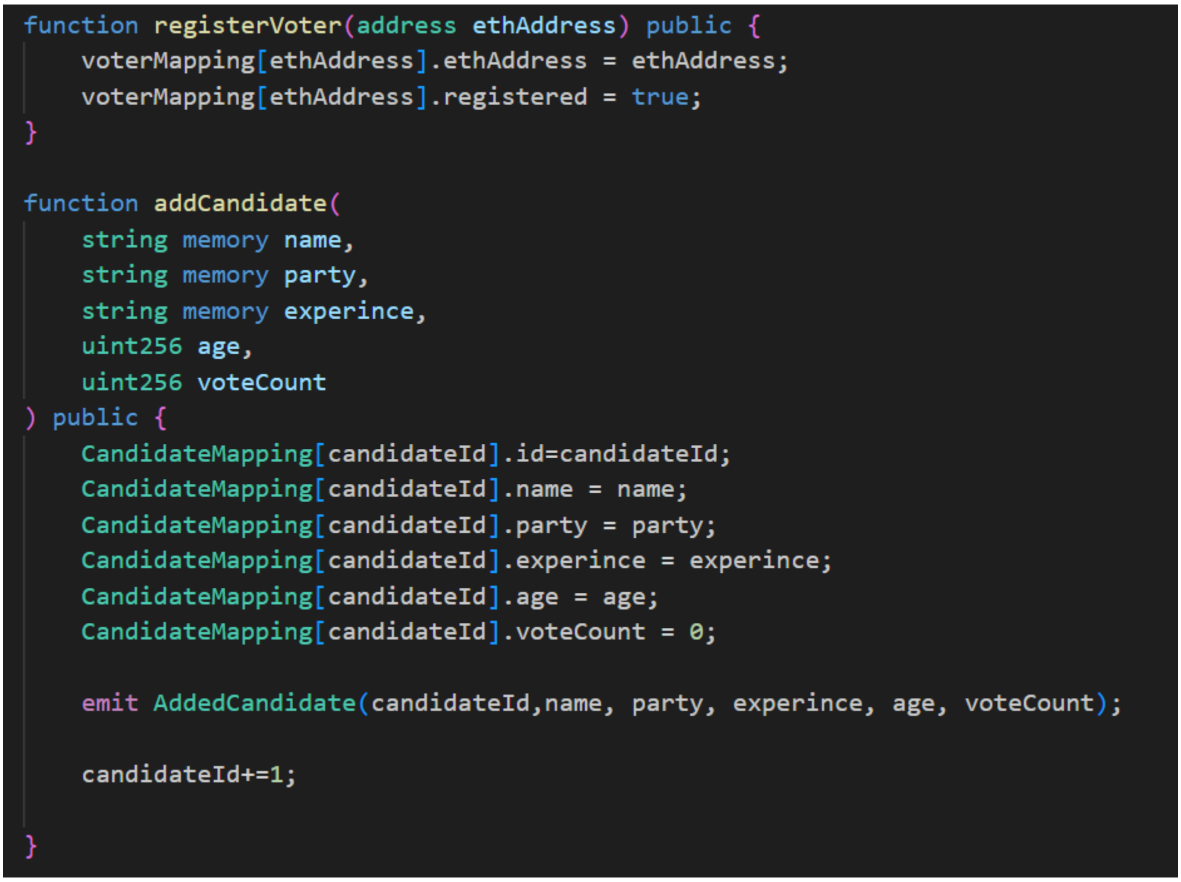
Other than the structs the voting contract also has Events, including addedCandidate, voted and phaseChanged. The first event added candidate has attributes like ID, name, party, experience, age and voteCount. This event is triggered when a new candidate is added by the admin. The next event which is voted has attributes including updated vote count of the candidate, voter which has voted for the candidate and candidateID of the candidate who has been voted for. This is triggered when a voter votes for a candidate. The vote count of that candidate increases and the address of the Voter which has voted for that candidate for that particular candidate ID is also recorded. The last event which is the phaseChange event happens when the election phase changes. There are three phases for an election, voting, result and registration. It has two attributes: the phase name and phase ID.



```
  event AddedCandidate(
    uint id,
    string name,
    string party,
    string experince,
    uint256 age,
    uint256 voteCount
);

event Voted(
    uint voteCount,
    address voter,
    uint candidateId
);

event PhaseChanged(
    string phase,
    uint phaseId

);
```

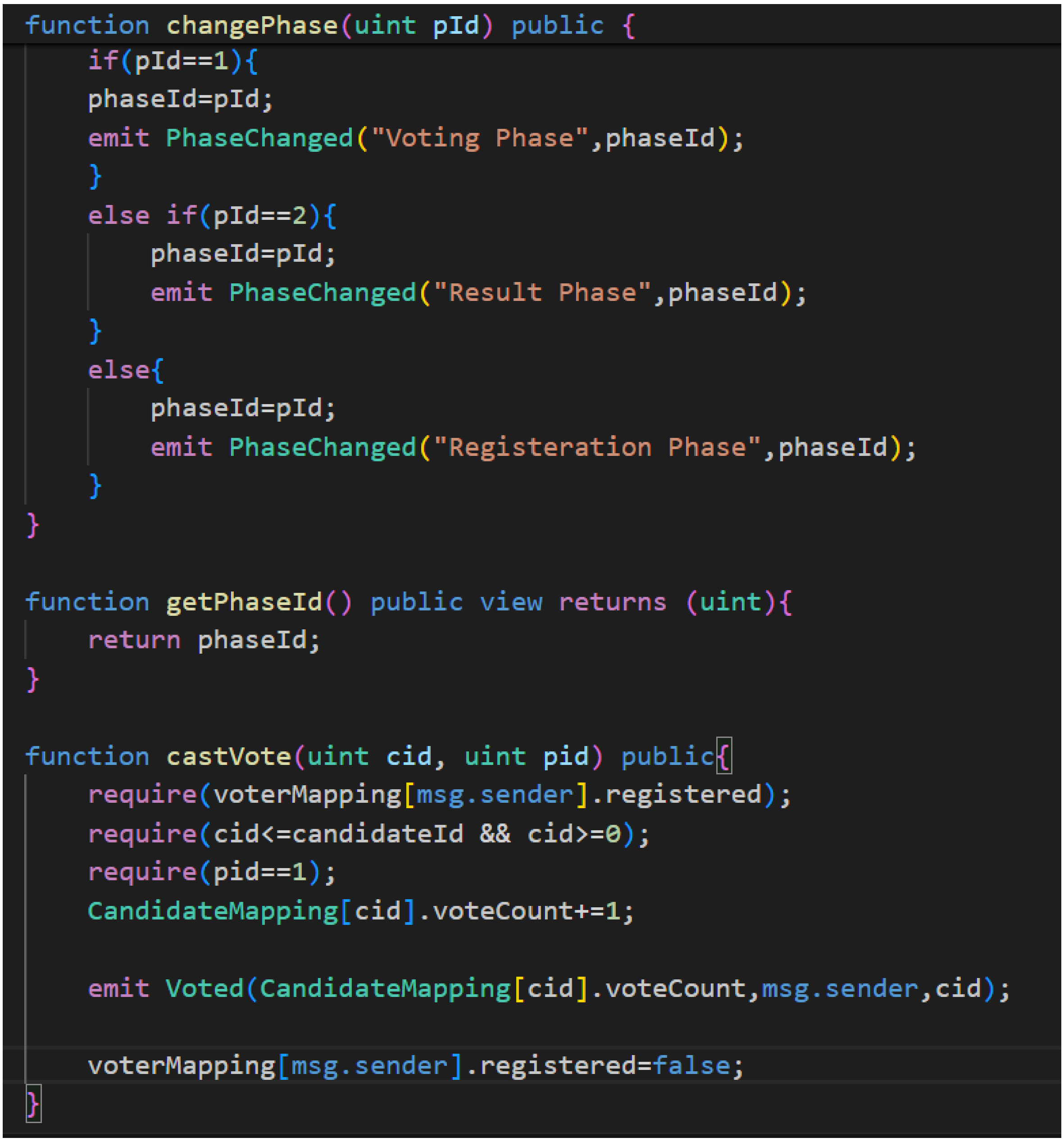
Figure 3.18. Events in the smart contract

And lastly, the smart contract has five functions which are: registerVoter, addCandidate, changePhase, getphaseid and castVote. All the functions have different functionalities. The registerVoter function takes the ethereum address of the Voter and registers the Voter in the database. The addCandidate function is triggered when the admin tries to add a new candidate. It takes the candidates name, party experience, age and current vote count. The vote count initially is set to 0. All of this is then added in the candidate mapping and the candidate ID is increased by one for every new candidate. The function changePhase takes the phase ID as an input and uses an if else statement to match the phase ID and change the phase accordingly. The function get phaseID returns the phaseID of the election and tells us which phase is going on right now. The last function which is the castVote function takes the candidate ID and the phase ID and adds the specific vote in the Voter mapping and increases the vote count of that particular candidate ID by one.



```
function registerVoter(address ethAddress) public {
    voterMapping[ethAddress].ethAddress = ethAddress;
    voterMapping[ethAddress].registered = true;
}

function addCandidate(
    string memory name,
    string memory party,
    string memory experince,
    uint256 age,
    uint256 voteCount
) public {
    CandidateMapping[candidateId].id=candidateId;
    CandidateMapping[candidateId].name = name;
    CandidateMapping[candidateId].party = party;
    CandidateMapping[candidateId].experince = experince;
    CandidateMapping[candidateId].age = age;
    CandidateMapping[candidateId].voteCount = 0;

    emit AddedCandidate(candidateId,name, party, experince, age, voteCount);

    candidateId+=1;

}
```

Figure 3.19. Functions in the smart contract (1)



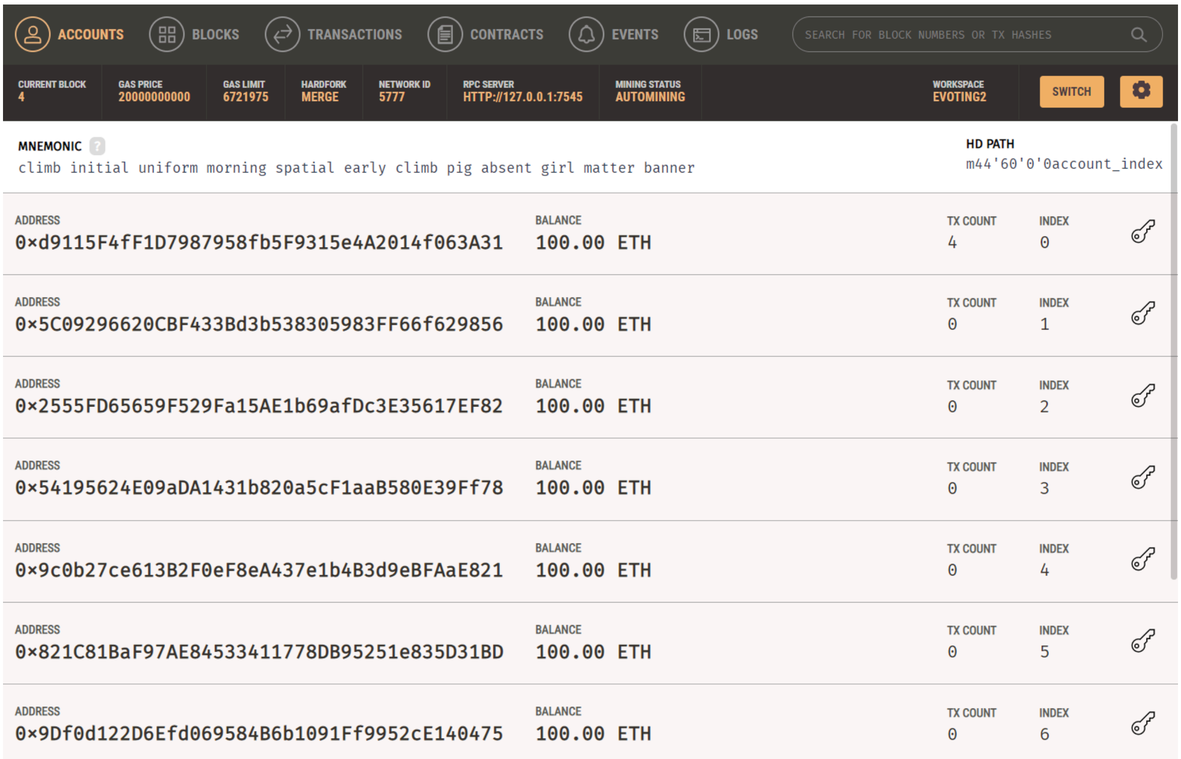
```
function changePhase(uint pId) public {
    if(pId==1){
    phaseId=pId;
    emit PhaseChanged("Voting Phase",phaseId);
    }
    else if(pId==2){
        phaseId=pId;
        emit PhaseChanged("Result Phase",phaseId);
    }
    else{
        phaseId=pId;
        emit PhaseChanged("Registeration Phase",phaseId);
    }
}

function getPhaseId() public view returns (uint){
    return phaseId;
}

function castVote(uint cid, uint pid) public{
    require(voterMapping[msg.sender].registered);
    require(cid<=candidateId && cid>=0);
    require(pid==1);
    CandidateMapping[cid].voteCount+=1;

    emit Voted(CandidateMapping[cid].voteCount,msg.sender,cid);

    voterMapping[msg.sender].registered=false;
}
```

Figure 3.20. Functions in the smart contract (2)

Setting Up Ganache:

Ganache is a personal blockchain tool used for Ethereum blockchain development that we have used to deploy smart contracts written in solidity programming language. It gives us a local development environment with some test accounts with 100 test ethers and allows us to simulate blockchain server without spending real ethers.

For using ganache we first make a new project and navigate our truffle-config.js file to connect vscode to ganache. This will run a ganache blockchain with 100 test ethers for the accounts.



CURRENT BLOCK 4 | GAS PRICE 20000000000 | GAS LIMIT 6721975 | HARDFORK MERGE | NETWORK ID 5777 | RPC SERVER HTTP://127.0.0.1:7545 | MINING STATUS AUTOMINING | WORKSPACE EVOTING2

MNEMONIC
climb initial uniform morning spatial early climb pig absent girl matter banner

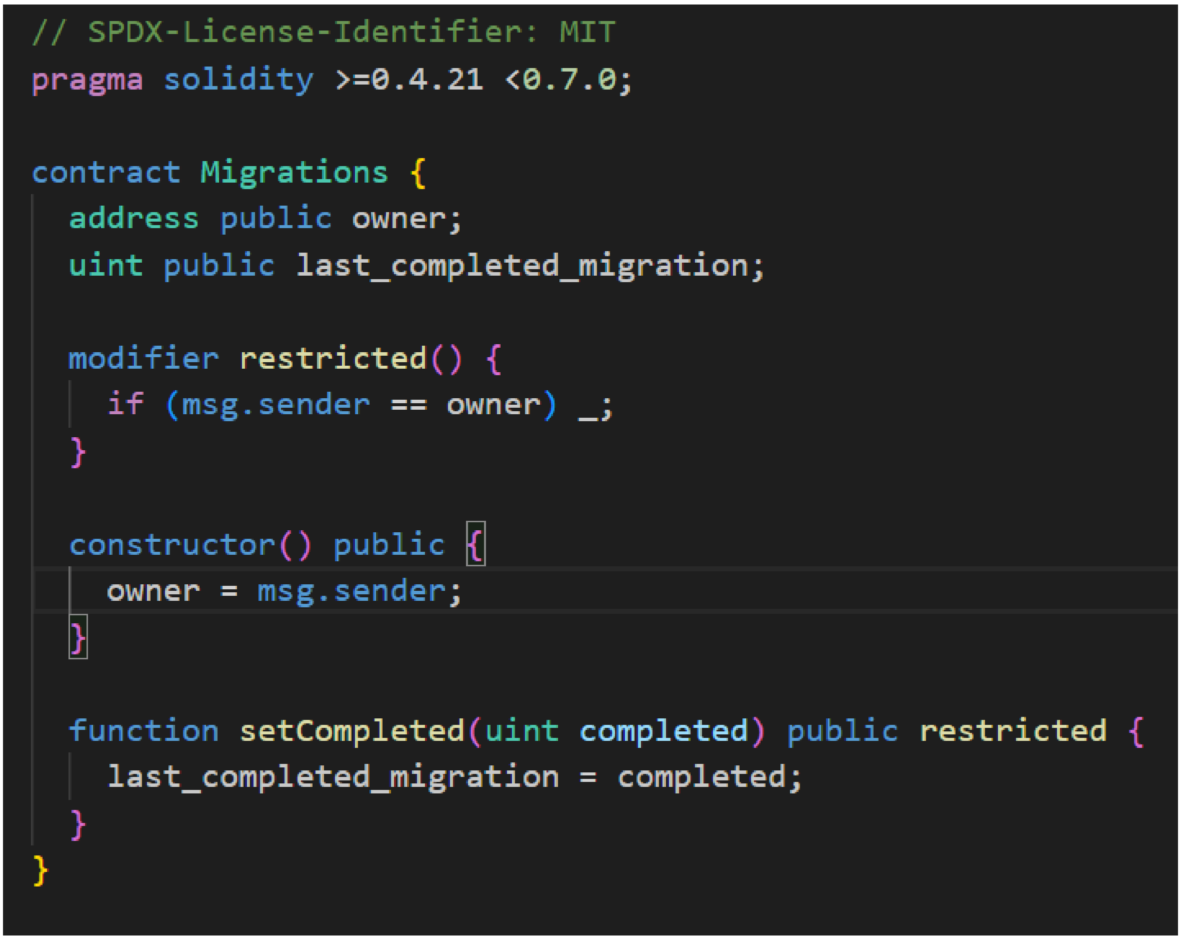
HD PATH
m44'60'0'0account_index

ADDRESS	BALANCE	TX COUNT	INDEX
0×d9115F4fF1D7987958fb5F9315e4A2014f063A31	100.00 ETH	4	0
0×5C09296620CBF433Bd3b538305983FF66f629856	100.00 ETH	0	1
0×2555FD65659F529Fa15AE1b69afDc3E35617EF82	100.00 ETH	0	2
0×54195624E09aDA1431b820a5cF1aaB580E39Ff78	100.00 ETH	0	3
0×9c0b27ce613B2F0eF8eA437e1b4B3d9eBFAaE821	100.00 ETH	0	4
0×821C81BaF97AE84533411778DB95251e835D31BD	100.00 ETH	0	5
0×9Df0d122D6Efd069584B6b1091Ff9952cE140475	100.00 ETH	0	6

Figure 3.21. Running Ganache server

Smart Contract Migration:

Smart contract migration refers to the process of deploying smart contracts on the blockchain network. When we write a smart contract in solidity programming language, it first has to be compiled into bytecode, and then it is deployed to the blockchain. Migration scripts are the ones that are used to automatically deploy the smart contracts on the blockchain. Figure (3.21) shows a solidity smart contract called "Migrations". This contract includes a state variable last_completed_migration that stores the ID of the last completed migration, along with the owner variable to show the address of the owner of the contract. The contract contains a constructor that initialises the owner variable with the address of the account that is deploying the smart contract onto the blockchain. The setCompleted function allows the owner of the contract which is the deploying account to update the last_completed_migration variable. In conclusion, this contract provides a basic structure for management of the migration tasks within the Ethereum blockchain.



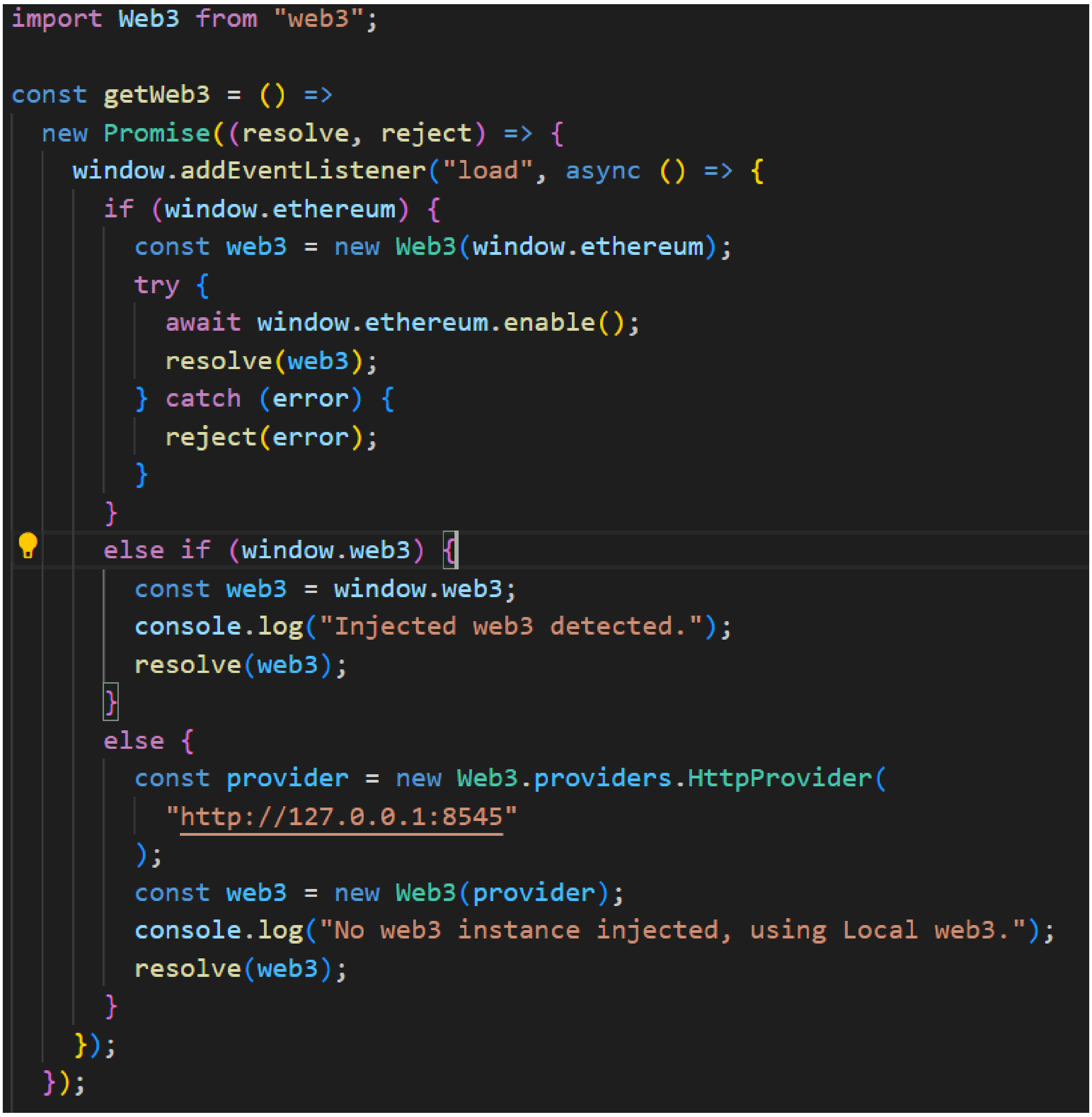
```
// SPDX-License-Identifier: MIT
pragma solidity >=0.4.21 <0.7.0;

contract Migrations {
  address public owner;
  uint public last_completed_migration;

  modifier restricted() {
    if (msg.sender == owner) _;
  }

  constructor() public {
    owner = msg.sender;
  }

  function setCompleted(uint completed) public restricted {
    last_completed_migration = completed;
  }
}
```

Figure 3.22. Migrations smart contract

Web3 Connection:

Web3.js is a JavaScript library that provides a way to communicate with the Ethereum blockchain. For using it we need to initialise a web3 instance and connect it to the ethereum network.

First make a promise that resolves with a web3 js instance or rejects it with any specific error provided by the developer. Then make an event listener for the loading event. When it loads, it checks if the window.ethereum object is there. If it is, it makes a Web3 instance. It then creates a new Web3 instance using the HTTP provider taking to the local Ganache blockchain running at http://127.0.0.1:8545.

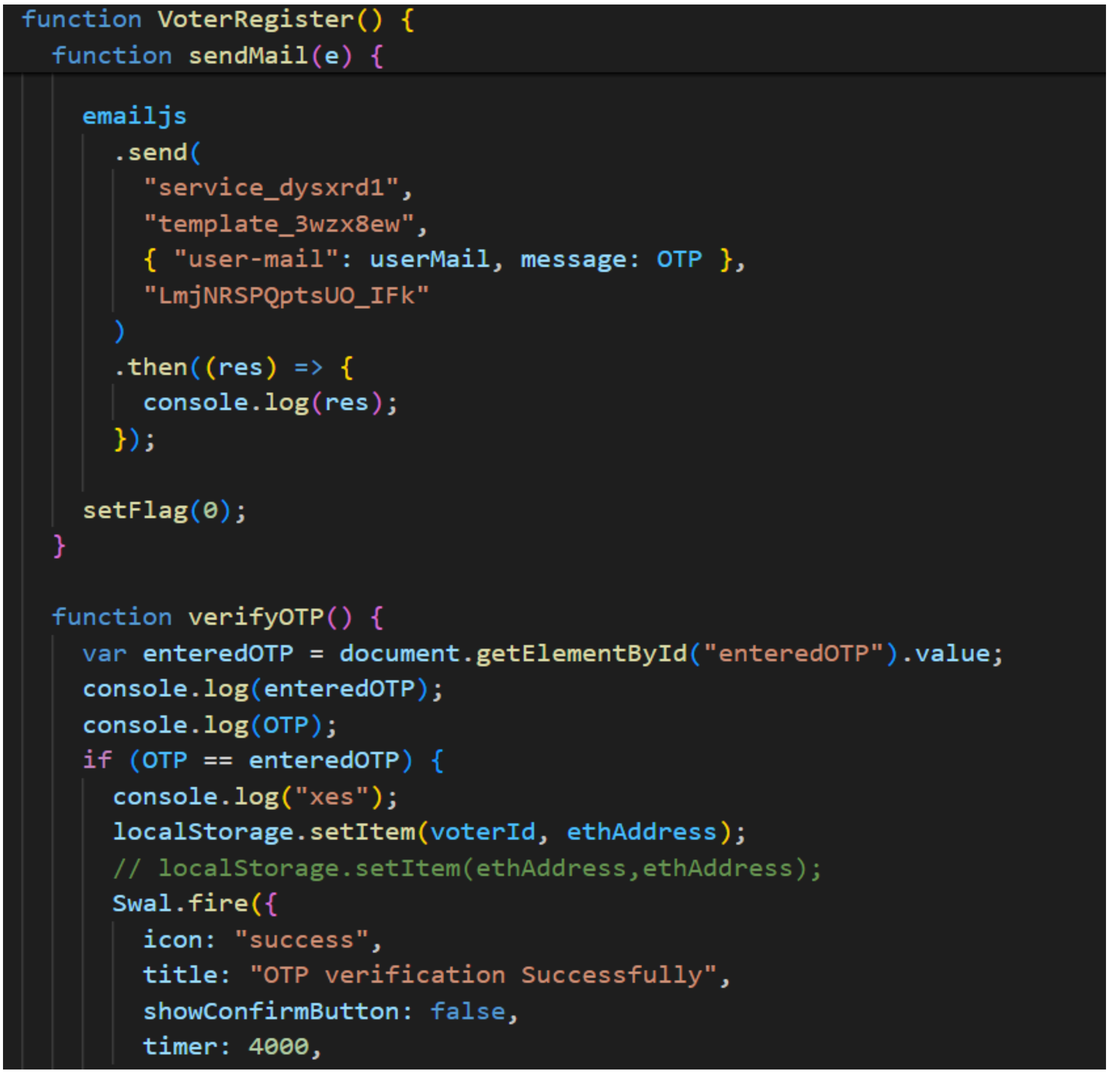


```
import Web3 from "web3";

const getWeb3 = () =>
  new Promise((resolve, reject) => {
    window.addEventListener("load", async () => {
      if (window.ethereum) {
        const web3 = new Web3(window.ethereum);
        try {
          await window.ethereum.enable();
          resolve(web3);
        } catch (error) {
          reject(error);
        }
      }
      else if (window.web3) {
        const web3 = window.web3;
        console.log("Injected web3 detected.");
        resolve(web3);
      }
      else {
        const provider = new Web3.providers.HttpProvider(
          "http://127.0.0.1:8545"
        );
        const web3 = new Web3(provider);
        console.log("No web3 instance injected, using Local web3.");
        resolve(web3);
      }
    });
  });
```

Figure 3.23. Migrations smart contract

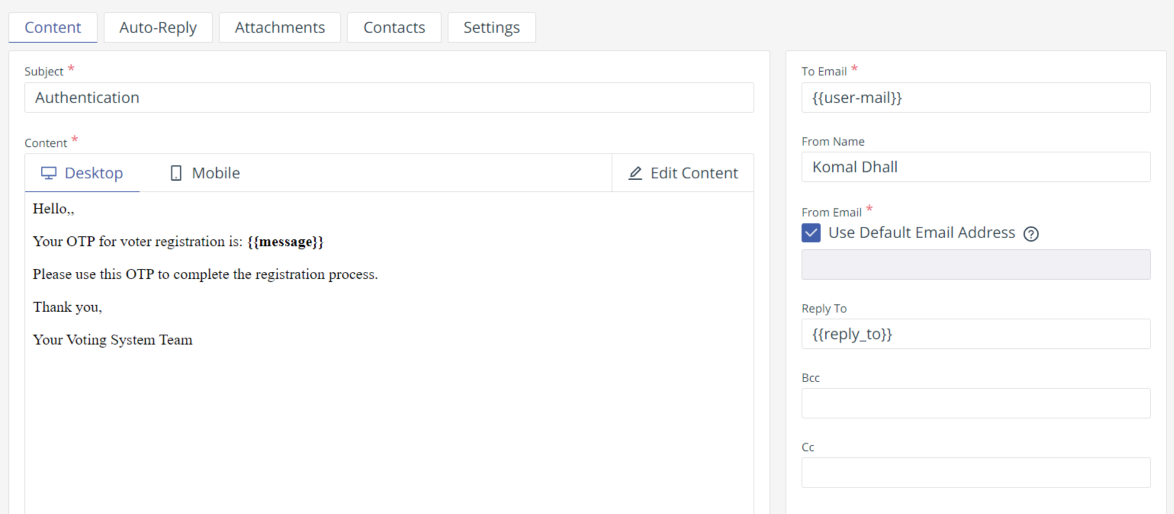
Voter Registration and OTP Verification:

The next step is to make the frontend and add the authentication functionalities. The VoterRegister function is responsible for the voter registration process. It uses state flag, OTP, voterId, and ethAddress variables to manage the registration of a voter. It generates a random OTP using digits from 0 to 9. When the user adds their registration details, an email containing the OTP is sent to the thor email address using the emailjs library. Usee then enter the received OTP for verification. If the entered OTP matches, the registration is successful, and the user's voter ID and Ethereum address are stored in the database.



```
function VoterRegister() {
  function sendMail(e) {

    emailjs
      .send(
        "service_dysxrd1",
        "template_3wzx8ew",
        { "user-mail": userMail, message: OTP },
        "LmjNRSPQptsUO_IFk"
      )
      .then((res) => {
        console.log(res);
      });

    setFlag(0);
  }

  function verifyOTP() {
    var enteredOTP = document.getElementById("enteredOTP").value;
    console.log(enteredOTP);
    console.log(OTP);
    if (OTP == enteredOTP) {
      console.log("xes");
      localStorage.setItem(voterId, ethAddress);
      // localStorage.setItem(ethAddress,ethAddress);
      Swal.fire({
        icon: "success",
        title: "OTP verification Successfully",
        showConfirmButton: false,
        timer: 4000,
```

Figure 3.24. Voter Registration



Subject *
Authentication

Content *

Hello,,

Your OTP for voter registration is: **{{message}}**

Please use this OTP to complete the registration process.

Thank you,

Your Voting System Team

To Email *
{{user-mail}}

From Name
Komal Dhall

From Email *
☑ Use Default Email Address

Reply To
{{reply_to}}

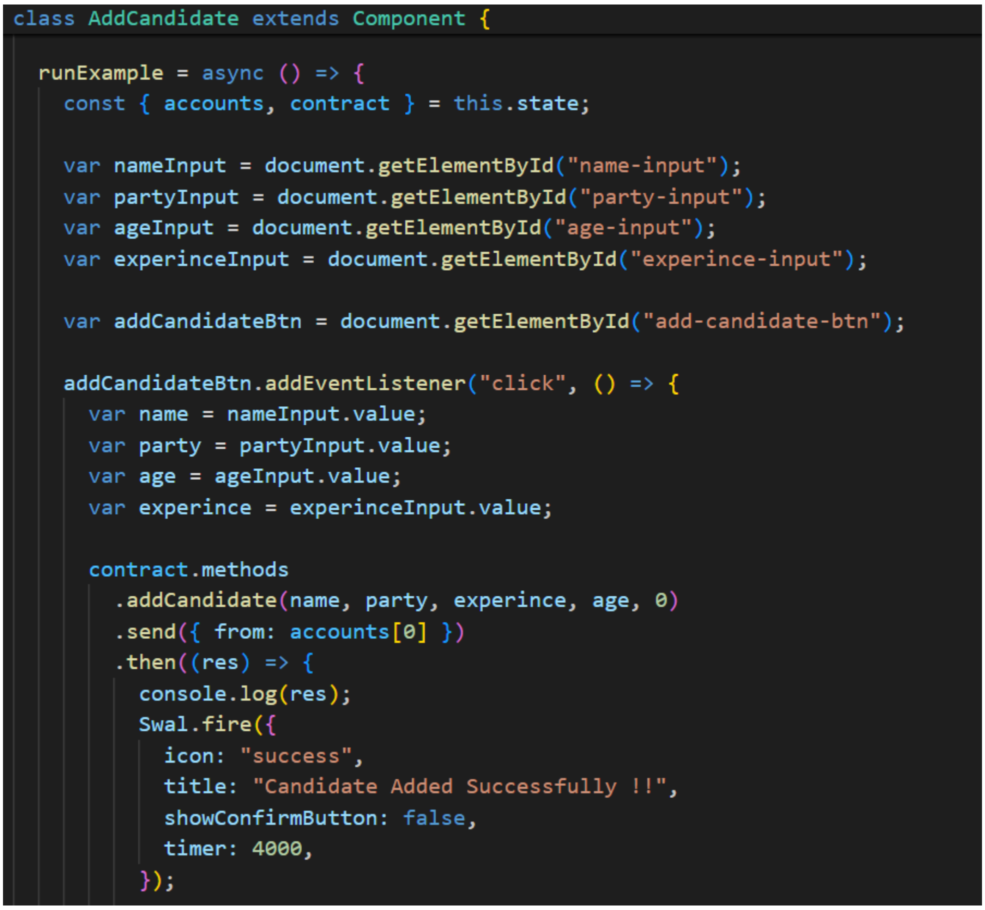
Bcc

Cc

Figure 3.24. EmailJS Template

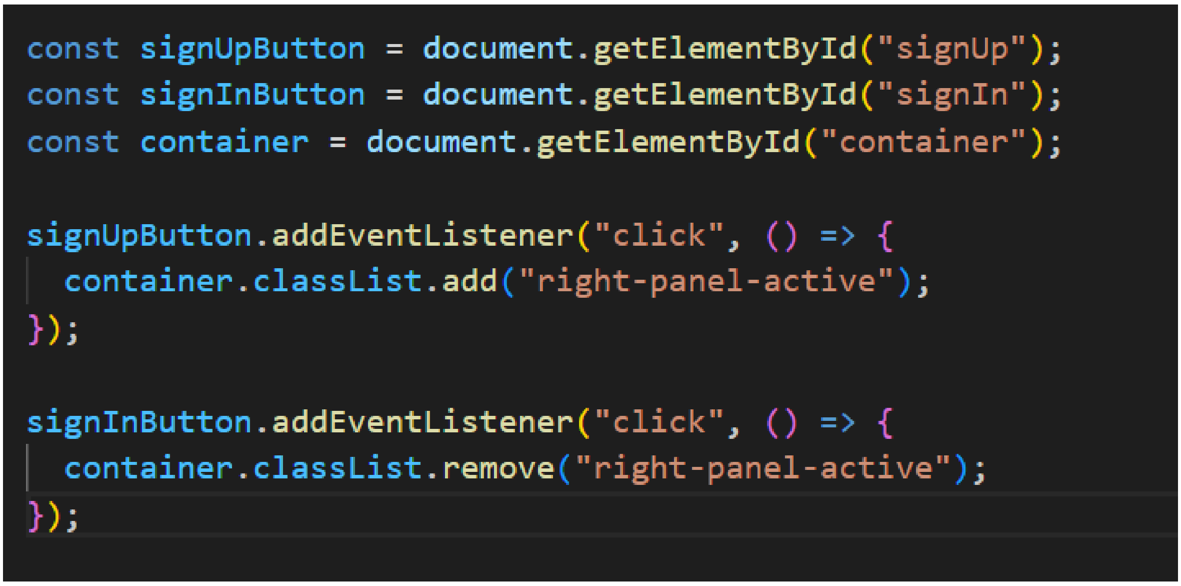
Similarly all other components of the website are built and connected to the ethereum network using web3 instance.

Other Code Snippets:



```
class AddCandidate extends Component {

  runExample = async () => {
    const { accounts, contract } = this.state;

    var nameInput = document.getElementById("name-input");
    var partyInput = document.getElementById("party-input");
    var ageInput = document.getElementById("age-input");
    var experinceInput = document.getElementById("experince-input");

    var addCandidateBtn = document.getElementById("add-candidate-btn");

    addCandidateBtn.addEventListener("click", () => {
      var name = nameInput.value;
      var party = partyInput.value;
      var age = ageInput.value;
      var experince = experinceInput.value;

      contract.methods
        .addCandidate(name, party, experince, age, 0)
        .send({ from: accounts[0] })
        .then((res) => {
          console.log(res);
          Swal.fire({
            icon: "success",
            title: "Candidate Added Successfully !!",
            showConfirmButton: false,
            timer: 4000,
          });
```

Figure 3.24. Add Candidate by Admin



```
const signUpButton = document.getElementById("signUp");
const signInButton = document.getElementById("signIn");
const container = document.getElementById("container");

signUpButton.addEventListener("click", () => {
  container.classList.add("right-panel-active");
});

signInButton.addEventListener("click", () => {
  container.classList.remove("right-panel-active");
});
```

Figure 3.26. Login

3.3.2 Tools and Technologies used in the Development Process

In order to develop an e-voting system using blockchain technology, various tools and technologies were used. The following tools were used in the development process:

Git - Git is a very popular platform for handling versions and tracking code changes among several collaborators. The project's repository used Git as a version control system that helped in handling the codebase during this project.

Visual Studio Code - This is an open source code editor from Microsoft for the operating systems of windows, linux and macOS. Developers can benefit from various capabilities including syntax highlighting, debugging, smart code autocomplete, and incorporated Git integration amongst others. The VS Code has a huge collection of additions and different themes which make it highly personalised according to the preferences of any developer. The e-voting system was developed using it as the primary code editor.

Node.js - Node.js is an open source runtime based on Chrome's V8 javascript engines. This is fast, efficient, and widely used for building scalable network applications. Node.js was adopted in this study as the backend programming tool to create the online voting application.

Figma - Figma is a cloud-based interface design tool allowing designers to work together on web and mobile user interfaces or UI designs in practice. It has many components which are important in the creation of UI design. Figma was used in designing a user interface for e-voting in this project.

MySQL - MySQL is an open-source relational database management system (RDBMS) that is used for managing structured data. It is a popular choice for web applications and is famed for its reliability, scalability, and performance. MySQL uses SQL (Structured Query Language) for querying and managing data.

Truffle - Truffle is a popular development framework for Ethereum technology, which is a blockchain platform that supports the making of decentralised applications (DApps) and smart contracts. Truffle provides developers with tools to simplify the process of building, testing, and deploying smart contracts and DApps on the Ethereum blockchain.

HTML - HTML stands for HyperText Markup Language. It is the standard markup language used to create structured web pages.

CSS - CSS stands for Cascading Style Sheets. It is a style sheet language used to define the presentation and layout of HTML documents.

Javascript - JavaScript is a programming language primarily used for adding interactive and dynamic behaviour to web pages.

Blockchain - Blockchain is a decentralised distributed digital ledger technology that records transactions from multiple computers in a way that makes it immutable and secure.

SHA256 - SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic hash function that gives a 256-bit hash value from an input data of arbitrary size.

Ethereum - Ethereum is a decentralised and open-source blockchain platform that is used for the development and deployment of smart contracts and decentralised applications (DApps).

Web3 - Web3 is the next generation of the internet, where decentralised technologies such as blockchain and cryptocurrencies play a big role in enabling peer-to-peer interactions and decentralised applications (DApps).

Ganache - Ganache is a developmental personal blockchain that allows testing of applications and smart contracts locally by an app or contract developer. The e-voting system prototype uses Ganache as a developmental blockchain to test and troubleshoot the associated smart contracts.

Remix - Remix is an Ethereum online development environment for smart contracts. This enables developers to write, test in addition to deploying smart contracts without any local installation.

MetaMask - A browser extension that enables one to interrelate with the Ethereum blockchain in his/her web browser. It allows users to store their Ethereum and ERC-20 tokens in a wallet, and it also has an interface that can be used when interacting with the apps.

Solidity - Solidity, is a programming language which allows to write smart contracts on the Ethereum blockchain. Its syntax resembles that of Java and other scripting languages, which means it can be readily read and written by software developers.

3.4 KEY CHALLENGES

Nevertheless, there are several advantages of online voting but there is the issue of accessibility among other concerns. Here are the key challenges faced and their solutions worked on while working on the e-voting system based on blockchain technology.

3.4.1 SECURITY MEASURES AND SMART CONTRACT DEVELOPMENT

- Challenge: Designing strong smart contracts and putting in place safe measures to eliminate loopholes as well as shield against attacks towards the blockchain network.
- Solution: Perform comprehensive code auditing, adopt generic libraries, and constantly upgrade security measures for the most stringent safeguard.

3.4.2 SCALABILITY AND PERFORMANCE OPTIMIZATION

- Challenge: The scalability problem is the biggest threat for most of the existing blockchain networks to allow the processing of many votes within the shortest period with speed and efficiency.
- Solution: The use of modern methods of consensus, sharding, or the layer 2 solutions to scale network effectiveness and capacity.

3.4.3 REGULATORY COMPLIANCE AND LEGAL FRAMEWORKS

- Challenge: Making sure that developed system is in line with varying legal and regulatory obligations at different countries or regions.
- Solution: Working hand in hand with lawyers to understand and create a framework for appropriate rules, regulations, and policies that support the design and functioning of the system.

3.4.4 USER EXPERIENCE AND ACCESSIBILITY

- Challenge: Providing an intuitive and user friendly interface for voters of various skill levels including users with different abilities.
- Solution: User testing and feedback sessions to enable iteration of the interface with focus on inclusiveness and easy navigation.

3.4.5 VERIFICATION, AUDITABILITY, AND TRANSPARENCY

- Challenge: Voter authentication, vote auditability, security and privacy, and the ability to allow each individual voter’s identification through a simple computer program.
- Solution: Use of cryptography in safeguarding vote integrity and availing audit trails to the public.

3.4.6 INTEGRATION AND INTEROPERABILITY

- Challenge: Smooth incorporation of the e-voting system into existing electoral framework and interoperability with different platforms.
- Solution: Integration in that it utilises conventional APIs and protocols for seamless interoperability but still keeping in line with security protocols.

3.4.7 PUBLIC TRUST AND EDUCATION

- Challenge: Overcoming scepticism, building trust among the population in confidence and safety of e-voting systems.

- Solution: Education outreach programs, clear information about the technology, involvement of stakeholders to promote trust.

3.4.8 RESOURCE MANAGEMENT AND SKILL ACQUISITION

- Challenge: Managing the lack of funds as well as hiring people who are familiar with blockchain technology and information security.
- Solution: Effective management of available resources, working as a team across disciplines, continuous training on changing technologies and practices.

CHAPTER 4 : TESTING

4.1 TESTING STRATEGY

The various strategies that were used for testing are as follows:

- Unit Testing
- Integration Testing
- Validation Testing
- User Validation Testing

4.1.1 BACKEND TESTING

Voter Registration : First we make sure that voters are able to register using their credentials. Also ensure that the voter registration data is securely stored in the database. Lastly it is important to validate that the voter registration is properly linked to their blockchain address.

Candidate Registration : Ensure that admin is able to register candidates using valid information. Make sure that the candidates registration data is securely stored in the database. Additionally please ensure that the candidates registration is linked to their address.

Voting Process : Make sure that individuals who are eligible to vote have the ability to cast their votes for the candidates of their choice. Double check that the votes are securely and permanently recorded on the blockchain leaving no room for alteration.

Voting Counting: Make sure that the aggregation and counting of votes are accurate. Ensure that the process of counting votes is transparent and can be verified. Validate that the total number of votes matches the recorded votes, on the blockchain.

4.1.2 INTEGRATION TESTING

Voter Registration and Authentication: Make sure to test the connection between the voter registration procedure and the authentication system. Confirm that only individuals who have registered as voters are able to access the voting platform.

Voting Process and Blockchain Interaction: Examine how the voting process and the blockchain network interact with each other. We need to confirm that votes are accurately recorded and securely stored on the blockchain. It is crucial to ensure that votes cannot be tampered with or altered in any way.

4.2 TEST CASES AND OUTCOMES

4.2.1 BACKEND TESTING

Voter Registration : We try to register a new voter and authenticate the voter. For this first we need an admin user. Registration of voters was found successfully. Voter authentication is currently being done by admin annually and is found successfull. Any attempt to register or login with invalid credentials is declined.

Candidate Registration : Admin is able to successfully register candidates using valid information. Any attempts to register with invalid credentials is declined.

Voting Process : A valid voter is allowed to vote for a valid candidate and this transaction is successfully recorded in the blockchain. Same voter is not allowed to vote twice and this was made sure of.

Voting Counting: Total votes made by the users and votes recorded on blockchain should be the same and accurate. This testing was done manually and was found to be successful.

4.1.2 INTEGRATION TESTING

Voter Registration and Authentication: We try to register a new voter and authenticate the voter. For this first we need an admin user. Registration of voters was found successfully.

Voter authentication is currently being done by admin annually and is found successfull. Any attempt to register or login with invalid credentials is declined. Voter details were successfully stored in the database.

Voting Process and Blockchain Interaction: Every casted vote should have a corresponding blockchain transaction. This was successful. Every casted vote created a new block in the blockchain. Voting should not be allowed when there is no connectivity to blockchain. This, when done, throws a network error message and thus is handled successfully.

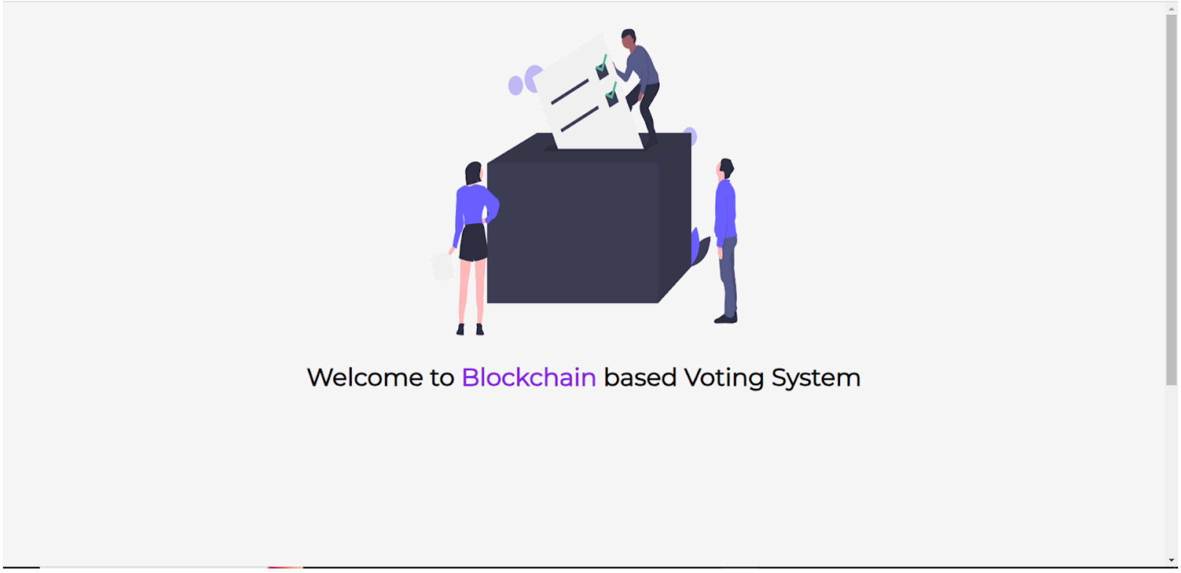
CHAPTER 5 : RESULTS AND EVALUATION

5.1 RESULTS

In this project, we were able to effectively use blockchain technology to create an electronic voting system and assess how well it functioned as a safe and convenient voting platform. The system incorporates voter verification for secure voting and has a web page accessible to voters and administrators alike. The votes are recorded on the blockchain, giving the election results an unchangeable and transparent record. The project's outcomes suggest that electoral fraud and manipulation concerns in India's democratic process can be successfully addressed by utilising blockchain technology. Voter involvement and engagement may rise as a result of the system's provision of a safe and convenient voting environment. The second advantage arising from the implementation of Aadhar verification is the assurance that only qualified voters can vote thus improving the overall security of the entire exercise. The transparency and non-manipulated nature of the results ensures that the system offers indisputable proof as to how the elections were conducted. This is important for enhancing confidence in the voting exercise and minimising protests on the outcome of the elections. Besides, the application of blockchain technology secures the votes, therefore safeguarding the integrity within the framework of the voting process.

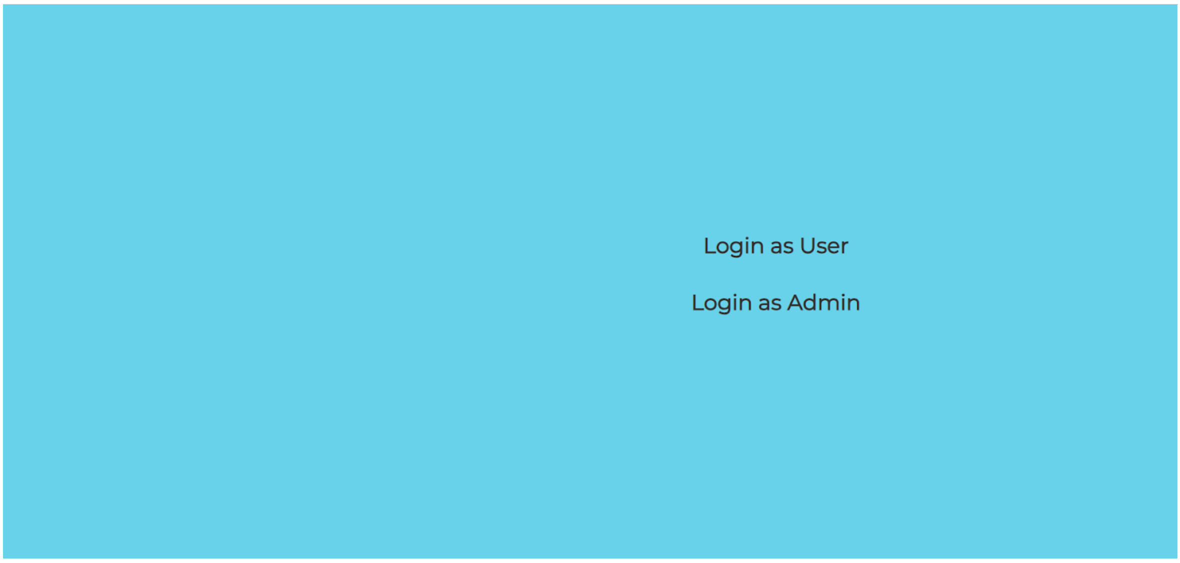
Nonetheless, the system also has its own disadvantages. Moreover, blockchain technology can be difficult to use and requires specific skills, thereby implying that it may not take on in some settings. Another problem with blockchain is that it is not yet scalable enough. In general, the outcomes of this project show that blockchain technology could serve as a reliable and safe medium for e-voting. Nevertheless, more studies are required to overcome the drawbacks of the system as well as investigating its actual usability in practice environments.

5.1.1 WEBSITE SCREENSHOTS



Welcome to Blockchain based Voting System

Figure 5.1. Landing Page

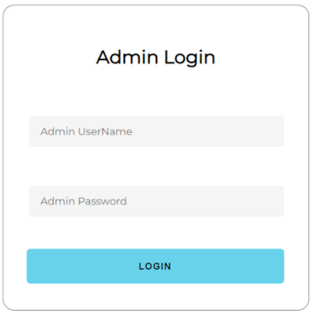


Login as User

Login as Admin

Figure 5.2. Login Selection Page

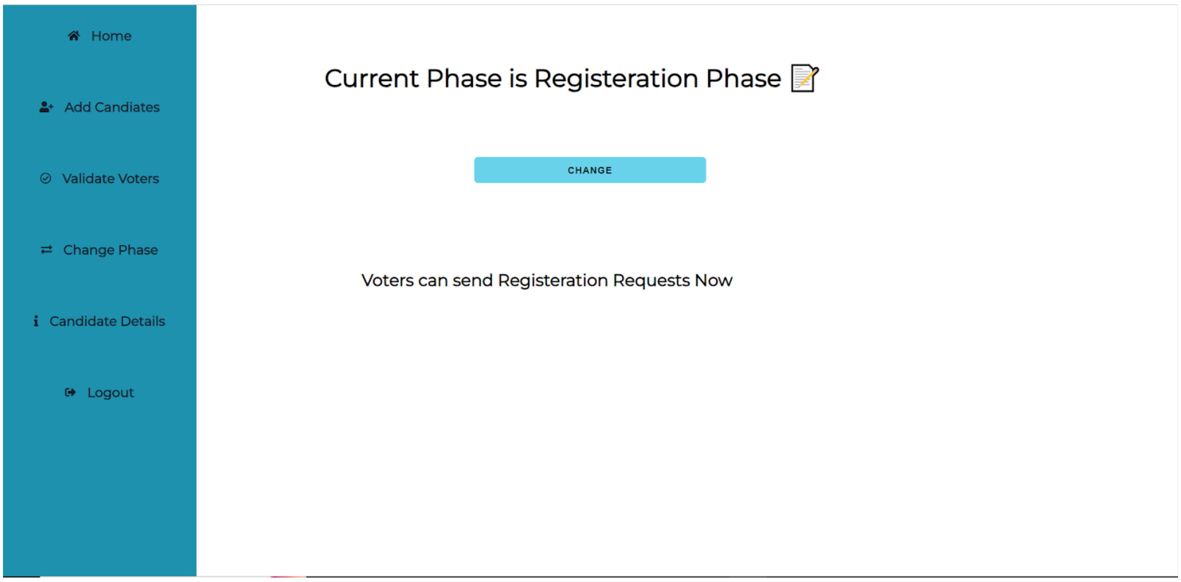
Admin Side Website


Admin Login

Admin UserName

Admin Password

Figure 5.3. Admin Login



Current Phase is Registeration Phase 📝

Voters can send Registeration Requests Now

Figure 5.4. Change Phase

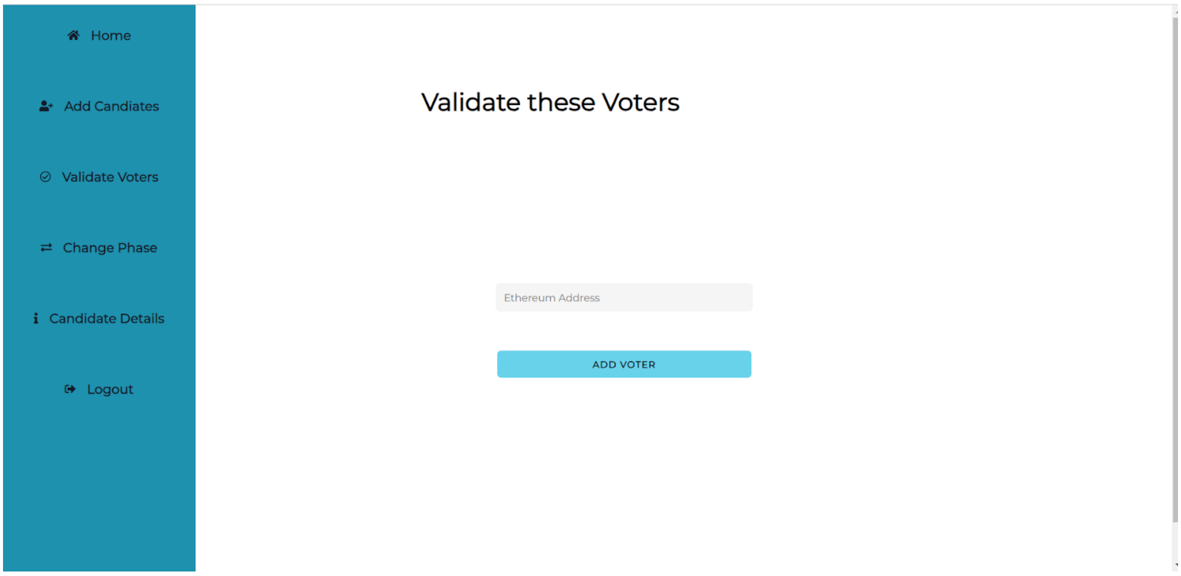


Figure 5.5. Validate Voter

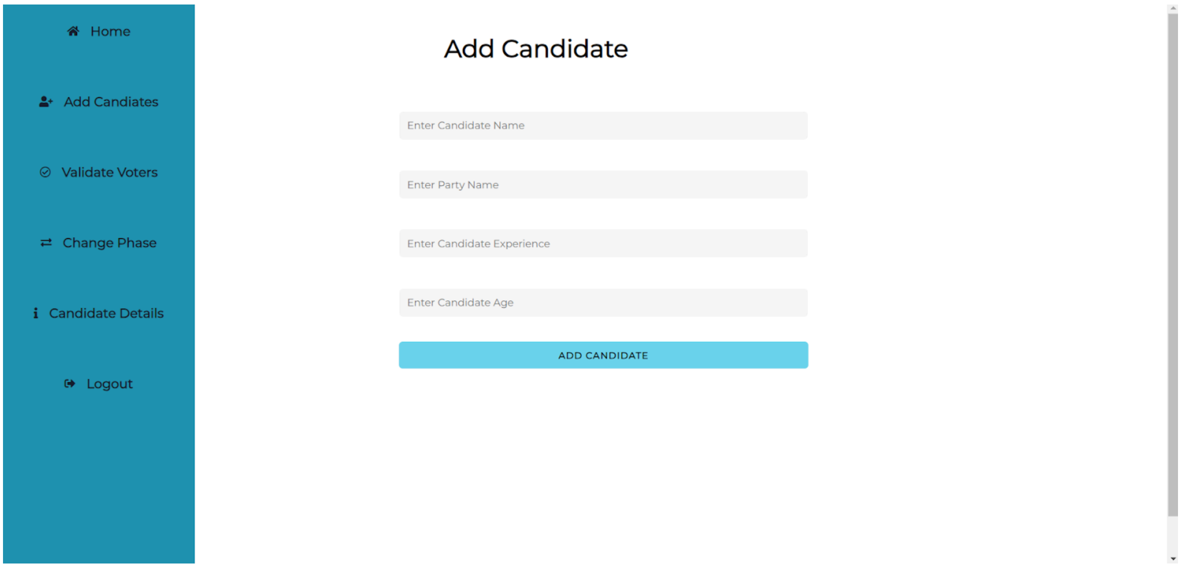
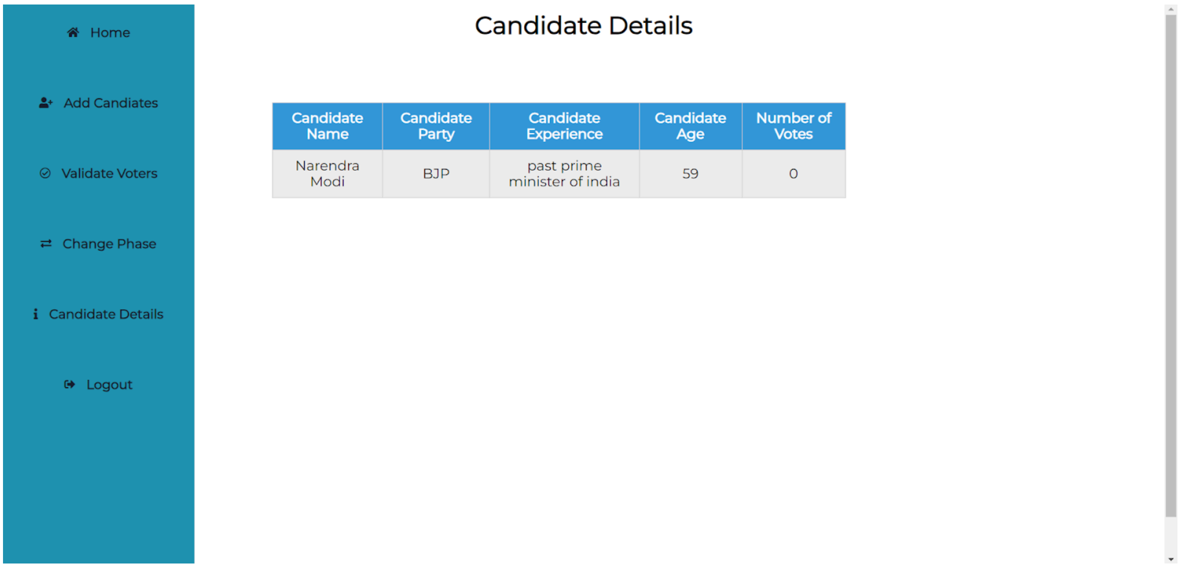


Figure 5.6. Add Candidate



Candidate Details

Candidate Name	Candidate Party	Candidate Experience	Candidate Age	Number of Votes
Narendra Modi	BJP	past prime minister of india	59	0

Figure 5.7. Candidate Details

Voter Side Website

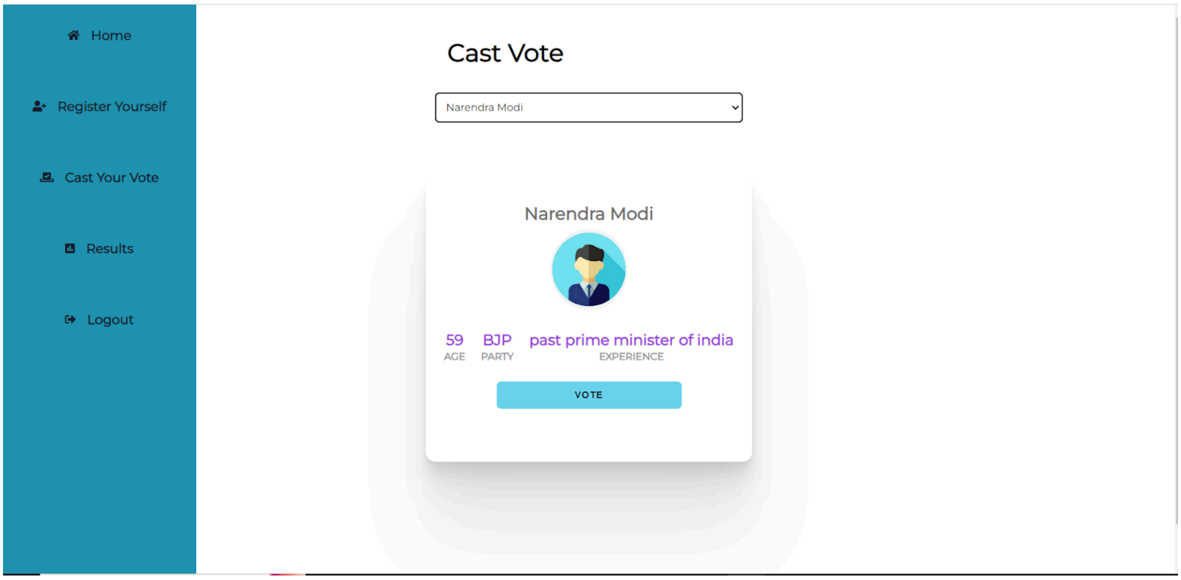


Figure 5.8. Vote Page

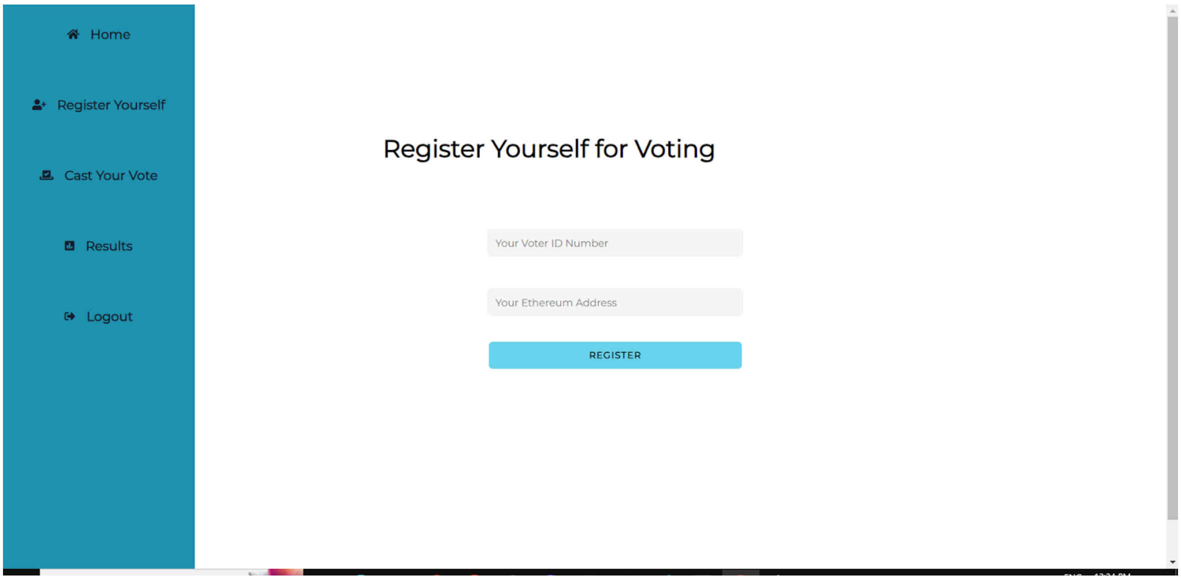
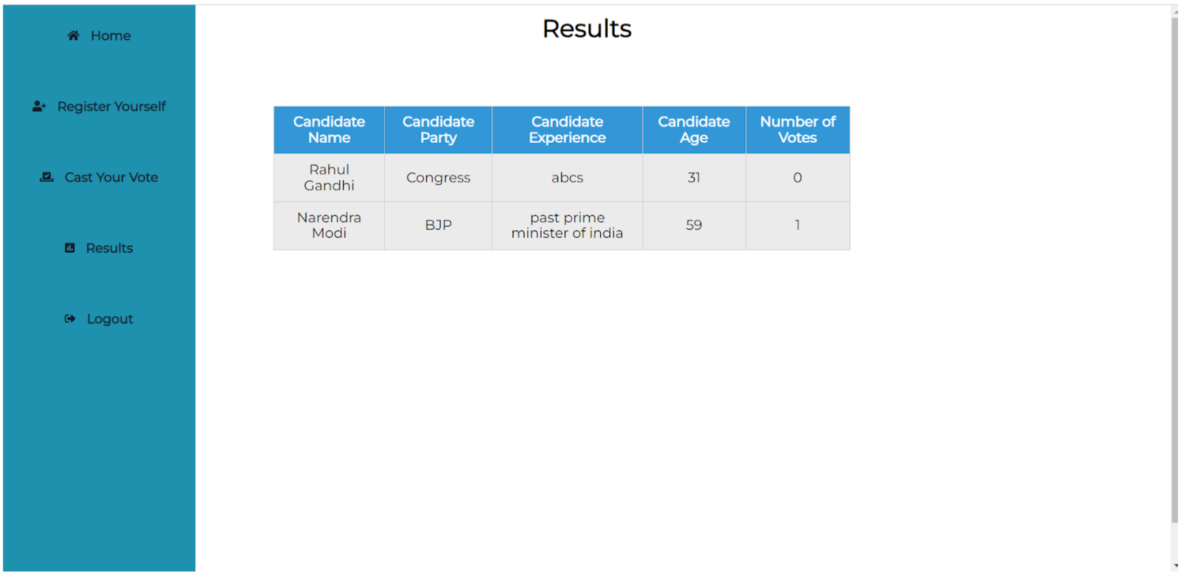


Figure 5.9. Voter Registration Page



Results

Candidate Name	Candidate Party	Candidate Experience	Candidate Age	Number of Votes
Rahul Gandhi	Congress	abcs	31	0
Narendra Modi	BJP	past prime minister of india	59	1

Figure 5.10. Result page

CHAPTER 6 : CONCLUSIONS AND FUTURE SCOPE

6.1 CONCLUSION

Our project provides a decentralised voting system which gives secure and transparent voting experience for voters and makes a difference compared to other voting applications with its highly secure blockchain technology. Because past transactions cannot be erased or reversed using distributed ledger technology in blockchain, security is increased because the database is visible to all users and the system is impervious to manipulation.

The provision of admin functionality improves voting security. The administrator is in charge of voter authentication and adding candidates to the election. Voter addresses can be used for authentication in the proposed system. The administrator is expected to be aware of which user addresses require authentication. Real-time viewing of the number of voted and unvoted people is also feasible. Due to a decrease in human error, operations are quicker and more dependable than with traditional voting methods. We used smart contracts in this application, which allows blockchain to be programmed. By establishing confidence between the voter and the programme, smart contracts remove the necessity for central authority. Small-scale elections can be conducted using the suggested application. Because blockchain technology is still developing and changing, there are some limitations to such projects like security improvements are needed before using blockchain-based voting apps on a wider scale.

Blockchain system also improves the voting process auditability and traceability, enabling participants to confirm the accuracy of the results. However, additional factors legal and regulatory constraints, user privacy protection constraints, and testing must be taken into account in order to ensure the application's dependance and compliance with industry standards. Regular updates and maintenance is also required to address many risks and adapt to changing environments.

6.1.1 LIMITATIONS

While the Minor Project has shown promising results and has several potential applications, it also has some limitations that should be considered. These include:

Limited scope: The e-voting system developed in this project has been tested in a controlled environment and with a limited number of participants. It may not be able to handle large-scale elections with millions of voters.

Security concerns: Though blockchain ensures utmost security, it cannot guarantee absolute safety levels. However, it means that there would also be potential vulnerabilities that can be exploited by malicious entities. They should ensure that they continuously monitor the system for possible security breaches.

Technical expertise: E-Voting system needs proficiency in block chain technology and website designing. However, small organisations and Local governments do not possess the finances or experience of managing such a system.

Voter engagement: Increased voter involvement and participation can be accomplished via the use of e-voting systems even though some people would be reluctant to utilise new technology and lack understanding of blockchain concepts. Education and training of voters on the voting system should be a major consideration as it builds confidence on its use.

6.1.2 APPLICATIONS AND CONTRIBUTIONS

The e-voting system from this minor project has various probable uses, especially with regard to democratic procedures. Some of the possible applications of the system include:

Government Elections: At local, state, and national level of administration, e-voting machines can be utilised for election purposes. The use of blockchain technology would ensure a safe and transparent system for the casting of votes, as it eliminates any possibility of electoral malpractice or manipulation.

Corporate Elections: This is especially true in most large corporations, where they have elections for a Board of directors and other critical positions. E-voting systems can be tailored to suit the requirements of the corporate elections, thus creating a reliable and useful environment for the organisation of such kinds of elections.

Non-Profit Organisations: Many non-profit organisations have to elect boards, positions or other offices. Given that most non-profits are low on resources, it may be very economical to use the e-voting system to hold free, fair and safe elections.

Educational Institutions: Many educational institutes, therefore, elect student councils or some other governing authorities through balloting exercises. Such e-voting can be adopted to enable safe and efficient electoral services in schools/educational institutions.

Online Voting: As people are becoming more dependent on internet-based devices and online interaction, there is an increased preference for online voting. It is important to note that e-voting is adjustable or modifiable, therefore it can be used to execute online voting while ensuring the security and efficiency of the process.

6.2 FUTURE SCOPE

Electronic voting systems face another difficult challenge: how will these technologies preserve one of the essential principles of any democratic election—the secret vote? Although the E-Voting system developed in this project is secure and efficient it still requires improvements in some points.

6.2.1 MOBILE APP

Another prospective study to undertake is in regard to designing a mobile phone application with which citizens could vote away from home. This will enable voting in any part of the world using people's phones or other mobile gadgets.

6.2.2 ENHANCED SECURITY FEATURES

In spite of the fact that the E-Voting is already conceived with security into consideration, there always is room for improvement. For instance, we could consider employing more sophisticated cryptography methods that would reinforce the voting information and bar any interference from external hackers. Moreover, they may focus on implementing stronger authenticity methods than only Aadhar number or any other peculiar citizenship identification.

6.2.3 SCALABILITY

Lastly, with increasing use of the system on large-scale elections, the system might need to be scaled up to deal with bigger volumes of web traffic and complicated vote situations. Using various cloud based hosting solutions can be explored to increase the scalability as well as make the system more flexible.

6.2.4 ACCESSIBILITY

The design of e-voting systems must provide equal access for every voter including the disabled. Therefore, future work should involve the provision of an accessible model of the E-voting system in accordance with the international standard of Web Content Accessibility Guidelines (WCAG 2.0) to facilitate participation by people with different kinds of abilities.

LIST OF PUBLICATIONS

[1] Taherdoost, H. “Smart Contracts in Blockchain Technology: A Critical Review” Information 2023, 14, 117.info14020117

[2] Jafar, U.; Aziz, M.J.A.; Shukur, Z. “Blockchain for Electronic Voting System—Review and Open Research Challenges” Sensors 2021, 21, 5874. https://doi.org/10.3390/ s21175874

[3] Khan, Kashif, Arshad, Junaid and Khan, Muhammad (2018) “Secure digital voting system based on blockchain technology” International Journal of Electronic Government Research (IJEGR), 14 (1). pp. 53-62. ISSN 1548-3886

[4] Michał Pawlak, Aneta Poniszewska-Marańda, Natalia Kryvinska, “Towards the intelligent agents for blockchain e-voting system” Procedia Computer Science, Volume 141, 2018, Pages 239-246, ISSN 1877-0509

[5] N. Kshetri and J. Voas, "Blockchain-Enabled E-Voting," in IEEE Software, vol. 35, no. 4, pp. 95-99, July/August 2018, doi: 10.1109/MS.2018.2801546.

[6] F. Þ. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa and G. Hjálmtýsson, "Blockchain-Based E-Voting System," 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), San Francisco, CA, USA, 2018, pp. 983-986, doi: 10.1109/CLOUD.2018.00151.

[7] F. Sheer Hardwick, A. Gioulis, R. Naeem Akram and K. Markantonakis, "E-Voting With Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy," 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 2018, pp. 1561-1567, doi: 10.1109/Cybermatics_2018.2018.00262.

[8] R. Hanifatunnisa and B. Rahardjo, "Blockchain based e-voting recording system design," 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), Lombok, Indonesia, 2017, pp. 1-6, doi: 10.1109/TSSA.2017.8272896.

[9] Ben Ayed, Ahmed. (2017). “A CONCEPTUAL SECURE BLOCKCHAIN-BASED ELECTRONIC VOTING SYSTEM” 10.5121/ijnsa.2017.9301.

[10] Nakamoto, S. (2008) Bitcoin: “A Peer-to-Peer Electronic Cash System” https://bitcoin.org/bitcoin.pdf

REFERENCES

[1] Nangula Shejavali. “Electronic Voting Machines”. In: Institute for Public Policy Research (IPPR) No 1 (2014).

[2] Jeannette Lynn Fraser. “THE EFFECTS OF VOTING SYSTEMS ON VOTER PARTICIPATION: PUNCH CARD VOTING SYSTEMS IN OHIO (MACHINES, ELECTION ADMINISTRATION, OVERVOTING, EQUIPMENT, BALLOT FORM)”. PhD thesis. The Ohio State University, 1985.

[3] Andrew W Appel, Richard A DeMillo, and Philip B Stark. “Ballot-marking devices cannot ensure the will of the voters”. In: Election Law Journal: Rules, Politics, and Policy 19.3 (2020), pp. 432–450.

[4] Baocheng Wang et al. “Large-scale Election Based On Blockchain”. In: Procedia Computer Science 129 (2018), pp. 234–237. issn: 1877-0509. doi: https://doi.org/10.1016/j.procs.2018.03.063. url: https://www.sciencedirect.com/science/article/pii/S1877050918302874.

[5] Lejun Zhang et al. “Secure and efficient data storage and sharing scheme for blockchain-based mobile-edge computing”. In: Transactions on Emerging Telecommunications Technologies (2021), e4315.

[6] Ori Jacobovitz. “Blockchain for identity management”. In: The Lynne and William Frankel Center for Computer Science Department of Computer Science. Ben-Gurion University, Beer Sheva (2016).

[7] Jennifer J Xu. “Are blockchains immune to all malicious attacks?” In: Financial Innovation 2.1 (2016), pp. 1–9.

[8] Jae Hyung Lee et al. “Systematic approach to analysing security and vulnerabilities of blockchain systems”. PhD thesis. Massachusetts Institute of Technology, 2019.

[9] Satoshi Nakamoto. “Bitcoin: A peer-to-peer electronic cash system”. In: Decentralised Business Review (2008), p. 21260.

[10] What is a block in the blockchain? https://medium.datadriveninvestor.com/what-is-a-block-in-the-blockchain-c7a420270373/.

[11] Blockchain Structure. https://www.oreilly.com/library/view/mastering-bitcoin/9781491902639/ch07.html/.

[12] Blockchain - Quick Guide. https://www.tutorialspoint.com/blockchain/blockchain_quick_guide.htm/.

[13] PoS explained. https://academy.binance.com/en/articles/proof- ofstake-explained/.

[14] Let’s learn more about the Proof of Stake. https://www.reddit.com/user/btcbamofficial/comments/oleo79/lets_learn_more_about_the_proof_ of_stake/.

[15] Simanta Sarmah. “Understanding Blockchain Technology”. In: 8 (Aug. 2018), pp. 23–29. doi: 10.5923/j.computer.20180802.02.

[16] Different types of blockchain technologies. https://thebossmagazine.com/different-types-of-blockchain-technologies/.

[17] What is Double Spending. https : / / corporatefinanceinstitute . com / resources/knowledge/other/double-spending/.

[18] Bitcoin logo. https://logos-world.net/bitcoin-logo/.

[19] What happens bitcoin after 21 million mined. https://www.investopedia.com/tech/what-happens-bitcoin-after-21-million-mined/.

[20] Gavin Wood et al. “Ethereum: A secure decentralised generalised transaction ledger”. In: Ethereum project yellow paper 151.2014 (2014), pp. 1–32.

[21] Ethereum docs - Account Types, Gas, and Transactions. https://ethdocs.org/en/latest/contracts-and-transactions/account-types-gas-andtransactions.html/.

[22] Shuai Wang et al. “Blockchain-enabled smart contracts: architecture, applications, and future trends”. In: IEEE Transactions on Systems, Man, and Cybernetics: Systems 49.11 (2019), pp. 2266–2277.

[23] What is Smart Contract. https : / / developers . rsk . co / guides / full - stack-dapp-on-rsk/part1-overview/.

[24] Introduction to Solidity. https://www.geeksforgeeks.org/introductionto-solidity/.

[25] NodeJS installation v14.17.5. https://nodejs.org/ko/blog/release/v14. 17.5/.

[26] Features Of Truffle Ethereum. https://www.edureka.co/blog/developingethereum-dapps-with-truffle/.

[27] What is Truffle Suite. https://www.upgrad.com/blog/what-is-trufflesuite/.

[28] Install Ganache. https://www.trufflesuite.com/ganache/.

[29] Web2 vs Web3. https://ethereum.org/en/developers/docs/web2- vsweb3/.

[30] MetaMask Chrome. https : / / chrome . google . com / webstore / detail / metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=en/.

BLOCKCHAIN

ORIGINALITY REPORT

10%	6%	4%	5%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Visvesvaraya Technological University, Belagavi Student Paper	<1%
2	www.coursehero.com Internet Source	<1%
3	howieliux.github.io Internet Source	<1%
4	www2.mdpi.com Internet Source	<1%
5	Anita A. Lahane, Junaid Patel, Talif Pathan, Prathmesh Potdar. "Blockchain technology based e-voting system", ITM Web of Conferences, 2020 Publication	<1%
6	Submitted to University of Hong Kong Student Paper	<1%
7	Submitted to AlHussein Technical University Student Paper	<1%

8	Hamed Taherdoost. "Smart Contracts in Blockchain Technology: A Critical Review", Information, 2023 Publication	<1 %
9	Submitted to Rose-Hulman Institute of Technology Student Paper	<1 %
10	www.ijraset.com Internet Source	<1 %
11	Submitted to 76830 Student Paper	<1 %
12	Submitted to National Institute of Technology Warangal Student Paper	<1 %
13	Submitted to mctrajiv Student Paper	<1 %
14	Po-Wei Chen, Bo-Sian Jiang, Chia-Hui Wang. "Blockchain-based payment collection supervision system using pervasive Bitcoin digital wallet", 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2017 Publication	<1 %
15	Submitted to Universiti Sains Islam Malaysia Student Paper	<1 %

16	Submitted to Loughborough University Student Paper	<1 %
17	link.springer.com Internet Source	<1 %
18	Michał Pawlak, Aneta Poniszewska-Marańda, Natalia Kryvinska. "Towards the intelligent agents for blockchain e-voting system", Procedia Computer Science, 2018 Publication	<1 %
19	www.agbiome.com Internet Source	<1 %
20	Submitted to kkwagh Student Paper	<1 %
21	scitepress.org Internet Source	<1 %
22	Submitted to University of Mauritius Student Paper	<1 %
23	export.arxiv.org Internet Source	<1 %
24	Submitted to Champlain College Student Paper	<1 %
25	Submitted to Napier University Student Paper	<1 %
26	Submitted to Taibah University Student Paper	<1 %

27	Submitted to University of Greenwich Student Paper	<1 %
28	medium.facilelogin.com Internet Source	<1 %
29	techwithfreezing.com Internet Source	<1 %
30	tools.namlabs.com Internet Source	<1 %
31	Submitted to Hacettepe University Student Paper	<1 %
32	dokumen.pub Internet Source	<1 %
33	hackernoon.com Internet Source	<1 %
34	www.matec-conferences.org Internet Source	<1 %
35	"A PRIMER ON BLOCKCHAIN", International Journal of Advances in Scientific Research and Engineering, 2018 Publication	<1 %
36	"Computational Science and Its Applications – ICCSA 2019", Springer Science and Business Media LLC, 2019 Publication	<1 %
	Submitted to Saint Leo University	

37	Student Paper	<1 %
38	Sang Ok Choi, Byung Cho Kim. "Voter Intention to Use E-Voting Technologies: Security, Technology Acceptance, Election Type, and Political Ideology", Journal of Information Technology & Politics, 2012 Publication	<1 %
39	Simona Ibba, Andrea Pinna, Maria Lunesu, Michele Marchesi, Roberto Tonelli. "Initial Coin Offerings and Agile Practices", Future Internet, 2018 Publication	<1 %
40	pure.royalholloway.ac.uk Internet Source	<1 %
41	"Decentralised Internet of Things", Springer Science and Business Media LLC, 2020 Publication	<1 %
42	Submitted to King's College Student Paper	<1 %
43	nijocet.fud.edu.ng Internet Source	<1 %
44	Ton Duc Thang University Publication	<1 %
45	Submitted to University of New South Wales Student Paper	<1 %

46	Submitted to University of Surrey Student Paper	<1 %
47	encyclopedia.pub Internet Source	<1 %
48	dev.to Internet Source	<1 %
49	Submitted to Graz University of Technology Student Paper	<1 %
50	Michał Pawlak, Aneta Poniszewska-Marańda. "Blockchain e-voting system with the use of intelligent agent approach", Proceedings of the 17th International Conference on Advances in Mobile Computing & Multimedia, 2019 Publication	<1 %
51	Submitted to University of Newcastle upon Tyne Student Paper	<1 %
52	cdr.lib.unc.edu Internet Source	<1 %
53	researchspace.ukzn.ac.za Internet Source	<1 %
54	technodocbox.com Internet Source	<1 %

55	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
56	micronanoeducation.org Internet Source	<1 %
57	researchrepository.wvu.edu Internet Source	<1 %
58	www.gecekitapligi.com Internet Source	<1 %
59	www.legalbusinessworld.com Internet Source	<1 %
60	www.sciencegate.app Internet Source	<1 %
61	"Leveraging Applications of Formal Methods, Verification and Validation. Industrial Practice", Springer Science and Business Media LLC, 2018 Publication	<1 %
62	"On the Move to Meaningful Internet Systems. OTM 2017 Conferences", Springer Science and Business Media LLC, 2017 Publication	<1 %
63	Debajani Mohanty. "Ethereum for Architects and Developers", Springer Science and Business Media LLC, 2018 Publication	<1 %

64	Submitted to Queensland University of Technology Student Paper	<1%
65	Rifa Hanifatunnisa, Budi Rahardjo. "Blockchain based e-voting recording system design", 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), 2017 Publication	<1%
66	Stijn Van Hijfte. "Blockchain Platforms", Springer Science and Business Media LLC, 2020 Publication	<1%
67	Submitted to University of Northampton Student Paper	<1%
68	eprajournals.com Internet Source	<1%
69	ijsret.com Internet Source	<1%
70	opus.lib.uts.edu.au Internet Source	<1%
71	univ45sby.ac.id Internet Source	<1%
72	vdocument.in Internet Source	<1%
	www.cubika.com.ar	

73	Internet Source	<1 %
74	"Innovative Mobile and Internet Services in Ubiquitous Computing", Springer Science and Business Media LLC, 2020 Publication	<1 %
75	"Intelligent Data Engineering and Automated Learning – IDEAL 2018", Springer Science and Business Media LLC, 2018 Publication	<1 %
76	Friorik P. Hjalmarsson, Gunnlaugur K. Hreioarsson, Mohammad Hamdaqa, Gisli Hjalmtysson. "Blockchain-Based E-Voting System", 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 2018 Publication	<1 %
77	Haibo Yi. "Securing e-voting based on blockchain in P2P network", EURASIP Journal on Wireless Communications and Networking, 2019 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

PLAGIARISM VERIFICATION REPORT

Date: ……………………….

Type of Document (Tick): PhD Thesis M.Tech/M.Sc. Dissertation B.Tech./B.Sc./BBA/Other

Name:______________________________Department:_________________Enrolment No _________

Contact No.______________________________E-mail.______________________________________

Name of the Supervisor: ____

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): _________________________

____

____

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

- Total No. of Pages =
- Total No. of Preliminary pages =
- Total No. of pages accommodate bibliography/references =

(Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found **Similarity Index** at................ (%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor) **Signature of HOD**

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Abstract & Chapters Details	
	• All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String		Word Counts	
			Character Counts	
Report Generated on		**Submission ID**	Page counts	
			File Size	

Checked by
Name & Signature **Librarian**

………

Please send your complete Thesis/Report in (PDF) & DOC (Word File) through your Supervisor/Guide at plagcheck.juit@gmail.com